

NETWORK AND DATA TRANSMISSION METHOD

Publication number: JP9093250 (A)

Publication date: 1997-04-04

Inventor(s): FUJIMORI JUNICHI; ABE TATSUTOSHI

Applicant(s): YAMAHA CORP

Classification:

- international: **H04L12/28; H04L12/40; H04L12/64; H04L29/06; H04L12/18; H04L12/28; H04L12/40; H04L12/64; H04L29/06; H04L12/18; (IPC1-7): H04L12/28**

- European: **H04L12/40F1; H04L12/40F2; H04L12/40F10; H04L12/64B; H04L29/06**

Application number: JP19950270738 19950926

Priority number(s): JP19950270738 19950926

Also published as:

JP3271493 (B2)

EP0766428 (A2)

EP0766428 (A3)

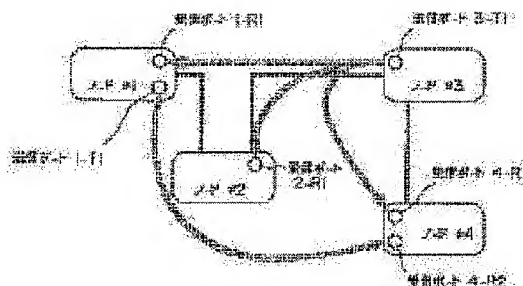
EP0766428 (B1)

US5825752 (A)

Abstract of JP 9093250 (A)

PROBLEM TO BE SOLVED: To attain isochronous transfer and multi-cast transmission on a network of a logic path interconnecting electronic devices.

SOLUTION: Nodes #1, #2, nodes #2, #3, and nodes #3, #4 are respectively interconnected by a physical cable. A prescribed band, a path for isochronous transfer using an isochronous channel number, and a path for multi-cast transfer using an allocated multi-cast channel number are set to the cable to transfer the data by this paths.



Data supplied from the esp@cenet database — Worldwide

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平9-93250

(43)公開日 平成9年(1997)4月4日

(51)Int.Cl.⁸

H 0 4 L 12/28

識別記号

庁内整理番号

F I

H 0 4 L 11/00

技術表示箇所

3 1 0 A

審査請求 未請求 請求項の数4 F D (全 31 頁)

(21)出願番号 特願平7-270738

(22)出願日 平成7年(1995)9月26日

(71)出願人 000004075

ヤマハ株式会社

静岡県浜松市中沢町10番1号

(72)発明者 藤森 潤一

静岡県浜松市中沢町10番1号 ヤマハ株式
会社内

(72)発明者 阿部 達利

静岡県浜松市中沢町10番1号 ヤマハ株式
会社内

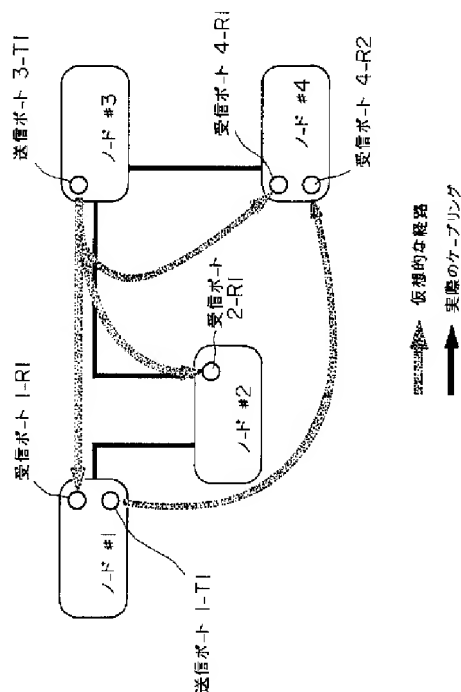
(74)代理人 弁理士 浅見 保男 (外1名)

(54)【発明の名称】 ネットワークおよびデータ伝送方法

(57)【要約】

【目的】電子機器間を接続する論理的パスのネットワーク上で、アイソクロナス転送とマルチキャスト転送を可能とする。

【構成】ノード#1とノード#2、ノード#2とノード#3、ノード#3とノード#4とは物理的なケーブルで接続される。このケーブル上に、所定の帯域とアイソクロナス・チャンネル番号とを用いるアイソクロナス転送のパスと、割り当てられたマルチキャスト・チャンネル番号を用いるマルチキャスト転送とのパスを設定して、このパスによりデータを転送する。



【特許請求の範囲】

【請求項1】 複数のノードと、該複数のノードが接続されると共に、該ノード間を接続する論理的なバスが構築されるバスにより構成されるネットワークにおいて、

前記ノードの各々は、少なくとも1つのポートを有し、該ポートのタイプとして、アイソクロナス・チャンネル番号と所定の帯域がバインドされたアイソクロナス・トーカー、アイソクロナス・チャンネル番号がバインドされたアイソクロナス・リスナー、マルチキャスト・チャンネル番号がバインドされたマルチキャスト・トーカー、およびマルチキャスト・チャンネル番号がバインドされたマルチキャスト・リスナーのタイプが少なくとも用意されていることを特徴とするネットワーク。

【請求項2】 前記バスに構築された論理的なバスがリセットされた場合、アイソクロナス・トーカー及びマルチキャスト・トーカーが、それぞれ獲得したリソース情報をトーカー情報報告パケットとしてブロードキャストし、各ノードはこのトーカー情報報告パケット、および受信側のノード上で保持されているバス情報に基づいて、前記論理的なバスを再構築することを特徴とする請求項1記載のネットワーク。

【請求項3】 トランスポート層が少なくとも、前記各ノードで獲得されたアイソクロナス・リソース情報と、該アイソクロナス・リソースのバインド情報を管理するアイソクロナス・マネージャと、前記各ノードで獲得されたマルチキャスト・リソース情報と、該マルチキャスト・リソース情報のバインド情報を管理するマルチキャスト・マネージャと、マルチキャスト転送やアイソクロナス転送を行うポート間に前記論理的なバスを設定するサービスを上位層に対して提供すると共に、設定されたバスの管理を行うバス情報管理（PIM）とを含むことを特徴とするネットワーク。

【請求項4】 複数のノードが接続されると共に、該ノード間を接続する論理的なバスが構築されるバスによりデータを伝送するデータ伝送方法において、1転送サイクルが、アイソクロナス・チャンネルを用いた所定の帯域のデータを転送するアイソクロナス転送と、マルチキャスト・チャンネルを用いたマルチキャスト転送からなり、転送すべきデータを、データの性質や前記チャンネルの空き状態に応じた前記アイソクロナス転送あるいは前記マルチキャスト転送により伝送することを特徴とするデータ伝送方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、複数のノード間がバス上に構築された論理的なバスにより接続されるネットワークおよびデータ伝送方法に関するものであり、特に、MIDI（Musical Instrument Digital Interfac

e）データやオーディオデータの伝送に適用して好適なものである。

【0002】

【従来の技術】 最近のAVシステムにおいては、多くの電子機器を関連させて接続することによりネットワークを構築するようにしているが、関連する電子機器間の接続は、同軸ケーブル、シールドケーブル、あるいは平行ライン等を用いて個別に行っている。これは、MIDIにおいても同様であって、MIDIにおいては楽器同士を同軸ケーブルやシールドケーブル等により個別に接続することにより、楽器のネットワークを構築している。

【0003】 このように、現在のMIDI（を含む電子音楽環境）では物理的な接続形態が機器間の経路となっており、この形態ではある場所でのネットワーク構成を別の場所で再現しようとした場合にユーザに負担を強いることになる。さらに、このようなネットワークにおいては、電子機器間を接続する接続線の数が多く、その占有場所をかなり必要とすること、および、一度はずしてしまうと元にもどす配線作業が大変であるという欠点がある。そこで、電子機器間の接続線を1本とし、この接続線上で電子機器間のデータのやり取りを行うようにしたネットワークが種々提案されている。

【0004】 その一例のネットワークにおけるプロトコル階層の概略を図35に示す。このプロトコル階層は、物理層102、リンク層103、トランザクション層104、およびシリアル・バス・マネージャ101からなっている。物理層102には、ノード間を接続する物理的なインターフェースが定義されており、電気的な信号とリンク層103が扱う論理シンボルとの変換が行われている。変換された電気的な信号は、さまざまなシリアル・バス・メディア上で転送される。この場合、物理層102はアービトレーションによって1ノードだけがデータを転送できるように制御されている。すなわち、同時に複数のノードからデータが転送されることはない。

【0005】 リンク層103は、ノードからノードへのアドレッシング、データの検査とフレーム化を行い、トランザクション層104に対して一方向のデータ転送サービスを提供している。データの転送は受信側からのACKによって確認される。また、後述するアイソクロナス転送方式による転送サービスも提供している。トランザクション層は、例えばIEEE 1212 CSR（Control and Status Register）アーキテクチャに基づいたノード間のトランザクション・サービス（要求-応答のプロトコル）を提供している。ただし、アイソクロナス・データに対してはいかなるサービスも行わない。シリアル・バス・マネージャ101は、各ノードの機能を表すCSRを管理するエンティティであり、アイソクロナス・チャンネルや帯域をローカルバス内で集中的に管理している。

【0006】 なお、アイソクロナス転送では、例えば1

25 μ secの周期でサイクルが設定されており、アイソクロナス・チャンネル番号を確保しているノードは、このサイクルに一回だけ確保した帯域分のパケット化したデータを転送することができる。また、このアイソクロナス転送は、特定のノードに対してパケットを転送するのではなく、アイソクロナス・チャンネル番号が付されたパケットが全ノード宛にブロードキャストされる。アイソクロナス転送をサポートしているそれぞれのノードは受信したすべてのアイソクロナス・パケットをリンク層103に通知する。リンク層103はパケットに付されたチャンネル番号を検出して転送されたデータを取り込むか否かを決定する。

【0007】このように構成されるネットワークは、例えば、IEEE P1394 高性能シリアルバス規格として提案されているが、図1に実線で示すように物理的にはノード間をケーブルで接続するだけで良く、このネットワークでアイソクロナス転送を行う場合、送信側はデータを転送するためのリソースとしてチャンネル番号と帯域を確保し、受信側は特定の送信ポートからのデータを受信するためにチャンネル番号を指定する。これにより、送信ポートからブロードキャストされたアイソクロナス・パケットを特定の受信ポートが受信できるようになる。受信すべきアイソクロナス・パケットか否かの判断は前記したようにリンク層103により行われ、受信されたパケットは上位層に渡される。

【0008】

【発明が解決しようとする課題】しかしながら、前記したアイソクロナス転送のように、常に伝送帯域を確保しておくネットワークにおいては、MIDIメッセージのように離散的なデータを転送する場合には、確保した伝送帯域が100%常時使用されるものとは限らないため有効に機能せず、ネットワークの使用効率を低下させるという問題点があった。そこで、本発明は離散的なデータを伝送する場合にも使用効率の低下しないネットワークおよびデータ伝送方法を提供することを目的としている。また、ネットワークの動作中に新たな電子機器の参入、あるいは離脱が行われても、ネットワークの再初期化や再構築を自動的に行うことのできるネットワークを提供することを他の目的としている。

【0009】

【課題を解決するための手段】前記目的を達成するために、本発明のネットワークは、複数のノードと、該複数のノードが接続されると共に、該ノード間を接続する論理的なバスが構築されるバスにより構成されるネットワークにおいて、前記ノードの各々は、少なくとも1つのポートを有し、該ポートのタイプとして、アイソクロナス・チャンネル番号と所定の帯域がバインドされたアイソクロナス・トーカー、アイソクロナス・チャンネル番号がバインドされたアイソクロナス・リスナー、マルチキャスト・チャンネル番号がバインドされたマルチキ

スト・トーカー、およびマルチキャスト・チャンネル番号がバインドされたマルチキャスト・リスナーのタイプが少なくとも用意されているようにしたものである。

【0010】また、前記ネットワークにおいて、前記バスに構築された論理的なバスがリセットされた場合、アイソクロナス・トーカー及びマルチキャスト・トーカーが、それぞれ獲得したリソース情報をトーカー情報報告パケットとしてブロードキャストし、各ノードはこのトーカー情報報告パケット、および受信側のノード上で保持されているバス情報に基づいて、前記論理的バスを再構築するようにしたものであり、さらに、トランスポート層が少なくとも、前記各ノードで獲得されたアイソクロナス・リソース情報と、該アイソクロナス・リソースのバインド情報を管理するアイソクロナス・マネージャと、前記各ノードで獲得されたマルチキャスト・リソース情報と、該マルチキャスト・リソース情報のバインド情報を管理するマルチキャストマネージャと、マルチキャスト転送やアイソクロナス転送を行うポート間に前記論理的なバスを設定するサービスを上位層に対して提供すると共に、設定されたバスの管理を行うバス情報管理(PIM)とを含むようにしたものである。

【0011】また、本発明のデータ伝送方法は、複数のノードが接続されると共に、該ノード間を接続する論理的なバスが構築されるバスによりデータを伝送するデータ伝送方法において、1転送サイクルが、アイソクロナス・チャンネルを用いた所定の帯域のデータを転送するアイソクロナス転送と、マルチキャスト・チャンネルを用いたマルチキャスト転送からなり、離散的に発生されるデータは、主に前記マルチキャスト転送により伝送されるようにしたものである。

【0012】本発明によれば、物理的な接続形態に依存しない仮想的な経路(論理的なバス)をネットワークの内部に構築することができる。すなわち、そのネットワークに接続されていれば、それぞれの機器の位置関係に関係なく2つの機器間に経路を設定し、その経路を使ってデータを交換することができる。これによって接続形態に関わらず仮想的な経路を確立することができるので、例えば機器の繋ぎ順を間違えることがなくデータを転送できないというミスがなくなる。また、論理的に接続されているため、物理的な接続を変更せずに機器間の接続を変更することができるようになる。

【0013】仮想的な経路のバス情報はネットワークに接続されている各機器がそれぞれ記憶する。そして、ネットワーク内の全てのバス情報を一括して別メディアに保存すれば、再構成を行なう場合でもそれをネットワークにロードすることによって素早く正確に構成を再現することができる。さらに、本発明はこのようなネットワークにおいて、所定の帯域を確保してデータを伝送することのできるアイソクロナス転送に加えて、転送データがある時に指定された複数のノードにデータの転送を行

えるマルチキャスト転送を行うこともできる。したがって、離散的なデータか否かにかかわらず、効率的な伝送を可能とすることができる。すなわち、リアルタイム性を要求されるMIDIデータやオーディオデータ等を効率よく伝送することができるようになる。

【0014】

【発明の実施の形態】本発明のネットワーク（musical Local Area Network：以下、mLANと記す）の説明を以下の順序で行なう。

1. mLANの概要
 1. 1 通信形態の概要
 1. 2 mLANプロトコル階層
 1. 3 mLANトランスポート層
 1. 4 データ転送時のアドレッシング
 1. 4. 1 パス
 1. 4. 2 PIM（パス情報管理）
2. mLANトランスポート層の仕様
 2. 1 mLANトランスポート層のサービス
 2. 1. 1 ポート
 2. 2 mLANトランスポート層の構成
 2. 3 アイソクロナス転送のサービス
 2. 3. 1 アイソクロナス・マネージャ
 2. 3. 2 アイソクロナス・リソースの確保とバインド処理
 2. 3. 3 アイソクロナス・リソース・バインド処理
 2. 3. 4 アイソクロナス送信データ受付処理、アイソクロナス送信処理
 2. 3. 5 アイソクロナス受信処理
 2. 3. 6 トーカー重複検出処理
 2. 3. 7 下位層とのコミュニケーション
 2. 3. 8 PIMとのコミュニケーション
 2. 3. 9 バスリセット対応処理、バスリセット完了通知受信処理
 2. 3. 10 トーカー情報変更処理
 2. 3. 11 NIMとのコミュニケーション
 2. 4 マルチキャスト転送のサービス
 2. 4. 1 マルチキャスト・マネージャ
 2. 4. 2 マルチキャスト・リソースの確保とバインド処理
 2. 4. 3 トーカー情報変更処理
 2. 4. 4 マルチキャスト送信処理
 2. 4. 5 マルチキャスト・リソース・バインド処理
 2. 4. 6 マルチキャスト受信処理
 2. 4. 7 下位層とのコミュニケーション
 2. 4. 8 PIMとのコミュニケーション
 2. 4. 9 NIMとのコミュニケーション
 2. 4. 10 バスリセット対応処理、バスリセット完了通知受信処理
 2. 5 PIM（パス情報管理）
 2. 5. 1 パスの設定処理

2. 5. 2 パスに対するデータ送信処理
2. 5. 3 パスの解放処理
2. 5. 4 バスリセット対応処理、トーカー情報変更処理
- 各種不成功通知処理
 2. 5. 5 NIMとのコミュニケーション
2. 6 NIM（ノード情報管理）
 2. 6. 1 ポート情報問い合わせ処理
 2. 6. 2 トーカー情報報告パケット処理
 2. 6. 3 下位層とのコミュニケーション
 2. 6. 4 バスリセット対応処理、バスリセット完了時処理
2. 7 mLANトランスポート層のファシリティ
 2. 7. 1 パス情報テーブル
 2. 7. 2 ノード情報テーブル
 2. 7. 3 マルチキャスト情報テーブル
 2. 7. 4 アイソクロナス情報テーブル
 2. 7. 5 ポート情報エントリー
 2. 7. 6 トーカー情報報告パケット
2. 8 mLANサイクル・ストラクチャー
2. 9 マルチキャスト転送とアイソクロナス転送との使い分け
2. 10 プラグ・アンド・プレイ

【0015】1. mLANの概要

1. 1 通信形態の概要
mLANの通信形態の概要を図1に示す。この図に示すように、mLANにおいては、ノード#1とノード#2、ノード#2とノード#3、ノード#3とノード#4とは物理的なケーブルにより接続されている。各ノードには送信ポートあるいは受信ポートというアクセスポイントが定義されている。このアクセスポイントは上位アプリケーションが直接アクセスすることができる最上位のものである。本発明のネットワークにおいては、このポート間に図示するような仮想的な経路（論理的パス）を確立し、この経路によりデータを転送できるようにしている。

【0016】例えば、図1に示すように仮想的な経路が設定された場合は、ノード#1からは送信ポート1-T1とノード#4の受信ポート4-R2との経路により、ノード#1からノード#4にデータを転送することができる。また、ノード#3の送信ポート3-T1は仮想的な経路によりノード#1の受信ポート1-R1、ノード#2の受信ポート2-R1、およびノード#4の受信ポート4-R1と接続されており、ノード#3からノード#1、#2、#4にこの経路により、データを転送することができる。

【0017】mLANにおいては、転送方式のひとつとしてアイソクロナス転送方式が定義されており、あらかじめ帯域を確保して時間的な遅延が保証されたリアルタイム性の高いデータ転送を行なうことが可能である。し

かしアイソクロナス転送方式は連続的に帯域を使用しているため、MIDI メッセージのように離散的なデータを転送する場合には有効に機能しない。そこで、mLAN ではアイソクロナス転送に加え非同期マルチキャスト転送という1対多の転送方式が新たに定義されている。アイソクロナス転送方式も1対多の転送を行なうことができ、mLANでは非同期転送のパケットについてもチャンネルという概念を導入している。

【0018】非同期マルチキャスト転送方式では送信元からブロードキャストされるパケットを受信側が選択して受信するような“受信側主導型”の転送を行なう。この方式は同じ内容のパケットを複数の送信先に転送する場合に、パケットを1つだけ送信すればよいので帯域を効果的に利用することができる。また、この方式ではデータを受信する選択権は受信側にあるので、例えば受信側のノード数が増えた場合でも送信側の負担は変わらない。このように、転送に関する処理の多くを受信側に行なわせることによって送信側の負担を少なくし、パケット送信におけるリアルタイム性を向上することができる。

【0019】1. 2 mLANプロトコル階層
次に、本発明のmLANのプロトコル階層の一例を図2に示す。この図ではmLANのプロトコル階層を、OSI (Open System Interconnection) 参照モデルと対応付けて示している。mLANのプロトコル階層は、下位層インフラストラクチャと上位層インフラストラクチャに大きく分けることができる。下位層インフラストラクチャはOSI参照モデルの第1層～第4層、上位層インフラストラクチャは第5層～第7層までに相当する。下位層インフラストラクチャではアプリケーション間でメッセージを交換するための手段が定義され、上位層インフラストラクチャではアプリケーション間で交換されるメッセージの内容が定義される。この下位層インフラストラクチャは転送経路を確立し、上位層に対して End-to-Endの転送サービスを提供する機能を有している。

【0020】mLANの上位層インフラストラクチャでは、転送される情報についての意味づけを行ない、それらの情報を利用するためのアプリケーション・サービスを定義する。この上位層はプレゼンテーション層(OSI第6層)26に相当するmLANプロトコル群11と、アプリケーション層(OSI第7層)27に対応するmLANアプリケーション12からなる。なお、セッション層(OSI第5層)25で実現される機能は、mLANトランスポート層10以下の階層で実現されるのでmLANはセッション層25に相当する階層は有していない。

【0021】mLANプロトコル群11では音楽情報の転送を効果的に行なうための付加的な制御メカニズムの定義、各種のデータ・フォーマットや転送プロトコルの定義を行なう。このプロトコルとしては、MIDI プロ

コル、Digital Audio プロトコル等が用意される。また、mLANアプリケーション12はmLANプロトコル群11で定義されたデータ交換方式を実際に利用するために構築されるアプリケーションである。

【0022】mLANの下位層インフラストラクチャは、トランスポート層(OSI第4層)24に相当するmLANトランスポート層10と、ネットワーク層(OSI第3層)23に相当するトランザクション層3と、データリンク層(OSI第2層)22に相当するリンク層2と、物理層(OSI第1層)21に相当する物理層1と、ノードコントローラ(Node Controller)4とから構成されている。mLANトランスポート層10は、ノード内にポートと呼ばれるサブアドレスが導入されて、ポート対ポートの通信が実現されると共に、下位層の規格の違いが吸収される。また、mLAN上位層に対して下位層の形式に依存しない共通の形式のサービス・インターフェイスを提供する。したがって、規格の異なる下位層が複数ある場合は、mLANトランスポート層10はそれぞれの下位層に対して設けられる。

【0023】mLANトランスポート層10内部には、PIM(パス情報管理)8、アイソクロナス・マネージャ(Isochronous Manager)7、マルチキャスト・マネージャ(Multicast Manager)6、トランザクション・マネージャ(Transaction Manager)5、およびNIM(ノード情報管理)9が定義されている。また、物理層1では、ノード間を接続する物理的なインターフェイスが定義されており、電気的な信号とリンク層が扱う論理シンボルとの変換が行なわれている。また、mLANではケーブル環境だけを使用するものとする。

【0024】さらに、リンク層2は、ノードからノードへのアドレッシング、データの検査とフレーム化を行ない、トランザクション層3に対して一方向のデータ転送サービスを提供している。データの転送は受信側からのACKによって確認される。また、リンク層2はアイソクロナス方式による転送サービスも提供している。トランザクション層3は、例えばIEEE1212 CSRアーキテクチャIEEE1212に基づいたノード間のトランザクション・サービス(要求-応答のプロトコル)を提供している。また、ノードコントローラ(シリアルバス管理に相当)4は各ノードの機能を表すCSRを管理するエンティティであり、アイソクロナス・チャンネル番号や帯域をローカルバス内で集中的に管理している。

【0025】1. 3 mLANトランスポート層
mLANトランスポート層10ではノード内にポートと呼ばれるサブアドレスを定義しており、このポート間でデータを交換するための転送サービスを提供する。また、mLANトランスポート層10はアイソクロナス転送、非同期パケット転送(トランザクション)に加えてマルチキャスト転送と呼ばれる1つの送信ポートから複数の受信ポートへのデータ転送プロトコルが定義されて

いる。

【0026】1. 4 データ転送時のアドレッシング
ポート間でマルチキャスト転送やアイソクロナス転送による転送を行なう場合には、ポートに対してチャンネルをバインドしてから転送を行わなければならない。送信側はデータを転送するためのリソースとしてチャンネル番号を確保し、受信側は特定の送信ポートからのデータを受信するためにチャンネル番号を指定する。チャンネル番号は転送を行なう際に各送信／受信ポートに対してダイナミックに割り当てられる資源であり、パスの初期化を行うバスリセットが行われた場合、その前後でポートと同じチャンネル番号が割り当てられるとは限らない。また、受信側ノードが送信側のノードを特定する場合にはノードIDが使われる。パスの初期化の際に各ノードに対して一意なノードIDが割り当てられる。したがって、SCSI (Small Computer System Interface) IDのようにユーザーが設定する必要がなくなり、また物理層1レベルでダイナミックな割り当てを行なうことによってローカルバス上でのノードIDの一意性が保証される。

【0027】このようにデータ転送のアドレッシングに用いられるノードIDとチャンネル番号はバスリセットによって再割り当てが行なわれ、それがバスリセット前後で異なる場合がある。そのため、アイソクロナス転送やマルチキャスト転送を行なっている時にバスリセットが発生するとアドレッシングが変更され、転送が継続できなくなる可能性がある。そこで、バスリセットをアプリケーション・ユーザーに意識させることなく、またバスリセットの前後でノードIDやチャンネル番号が変わった場合でも転送が継続できるように、mLANトランスポート層10ではバスという経路情報を保存し、このバス情報をもとに転送経路の再構成を行なうようにしている。

【0028】1. 4. 1 パス

また、アイソクロナス転送とマルチキャスト転送を行なうポートに設定される転送経路に関する情報であるバスは、コネクション・オリエンテッドなものではないと共に、複数の受信ポートにデータを転送するための経路情報である。この場合、データ転送自体はコネクションレスで行なわれる。バスは転送経路の受信側となるノード上に保存され、バスリセットやパワーリセットによってパスの初期化が行なわれた後にmLANトランスポート層10によって再設定が行なわれる。設定されたバス情報はmLANトランスポート層内のPIM (バス情報管理) 8と呼ばれるモジュールによって管理される。

【0029】なお、アイソクロナス転送やマルチキャスト転送では1 (送信側) 対多 (受信側) の転送方式を取るため、送信側でパスの再構成を行なうものだとすると送信側が行なわなければならない作業が多くなり、データ転送のリアルタイム性が損なわれる可能性がある。例え

ば、送信側が受信側のリストを保存していた場合、そのリストの中の受信ノードがバス上に見つからないと送信側がその状況を解決しなければならない。また、送信側には不特定多数の受信ノードへのバス情報を記憶する容量を確保するのが困難なノードが存在する。例としてマウス、キーボードなどの簡単な機構を持ったノードがあげられる。以上のような理由でバス情報は受信側ノード上で保持されている。

【0030】前記したようにバス情報はバスリセットによってノードIDやチャンネル番号が変更された場合に転送経路を再構築するために保持される情報である。したがって、バス情報として保持される情報はバスリセットによって変更されないパラメータでなければならない。そこでバス情報は、

- ・送信側ポートのポートID
- ・送信側ポートがあるノードの Node Unique ID
- ・受信側のポートID

という形で保持される。これを図示すると、図4に示すようなテーブルとなる。

【0031】なお、ポートIDはアプリケーションによって決定され、Node Unique ID は工場出荷時に製品ごとに異なるIDが設定されるのでバスリセットによって変更されることはない。mLANトランスポート層10は、バスリセットによって変更されるチャンネル番号やノードIDの変化に対しては内部的に情報更新を行ない、上位アプリケーションに対してはバスリセットによって変化しないリソースを提供する。これによってアプリケーションはバスリセットの発生に関係なく継続して転送を行なうことができる。

【0032】1. 4. 2 PIM (バス情報管理)

また、mLANトランスポート層10内に定義されているPIM (バス情報管理) 8は、ポート間に設定されるパスの管理を行なうモジュールであり、パスの設定／解放、設定されたバス情報の保持などを行なう。また、PIM 8はパスの動作中におけるノードの参入／離脱や電源再投入によって発生するバス初期化プロセス後に保持されていたバスを自動的に再構成する機能を有している。そして、PIM 8はマルチキャスト転送やアイソクロナス転送を行なうポート間にバスを設定するサービスを上位アプリケーションに提供し、このサービスによって設定されたバスは、図4に示すバス情報テーブルとしてPIMによって記憶される。

【0033】また、ノードの物理的な参入／離脱によってバスリセットが発生し、パスの再初期化が行なわれる場合、この再初期化のプロセスで各ノードに対して動的にノードIDが割り当てられるので、バスリセットの前後ではノードに割り当てられるノードIDが変更される可能性がある。ノードIDが変更されるとバスリセット前後でノードに対するアドレッシングが変更されることになるので正しい転送が行なえなくなる。PIM 8はこ

のようにバス構成に変更があった場合に、内部で保持しているバス情報テーブルを更新してバスリセット後も障害なくデータ転送が行なわれるようにする。これによって上位層はバスリセットによるバス構成の変更を意識する必要がなくなる。またPIM8が保持したバス情報は電源OFF後も保持されるので、電源再立ち上げ時にもバスは自動的に再設定される。

【0034】2. mLANTランスポート層の仕様

2. 1 mLANTランスポート層のサービス
mLANTランスポート層10は上位層に対して3種類の転送方式を提供する。3種類の転送方式は、マルチキャスト転送、アイソクロナス転送、およびトランザクション転送であり、それぞれの転送方式は以下のような特徴を持っている。

A. マルチキャスト転送

マルチキャスト転送は1つの送信用ポートから複数の受信用ポートに対して行なわれる非同期パケット転送である。この転送方式はmLANTランスポート層10で定義されている。マルチキャスト転送はブロードキャスト機能を利用するのでACKは返らない。したがって送信側が転送の成否を知ることができないが、送信するパケットは1つですむので帯域を効率的に使うことができる。

【0035】B. アイソクロナス転送

アイソクロナス転送方式はリンク層2で定義されているデータ転送である。送信遅延が保証されており、あらかじめ各ノードが帯域を確保してから転送を行なうのでネットワーク全体の負荷は管理されている。この転送方式は、定期的に一定量の転送要求が行なわれるデータあるいは厳密なタイミングが要求されるようなデータの転送に適している。

C. トランザクション

トランザクションはトランザクション層3で定義されている1対1の双方向通信である。コネクションレスではあるがACK/リトライ機能が実装されているので高品質のデータ転送を行うことが可能である。

【0036】2. 1. 1 ポート

次に、mLANTランスポート層10におけるサービス・アクセス・ポイントであるポートについて説明する。上位層はポートに対してアクセスすることによってmLANTランスポート層10が提供するサービスを利用することができる。前述したような各種方式によるデータ転送は特定の属性を持つポート間でのみ可能とされる。すなわち、異なる属性を持つポートに対して転送を行なうことはできない。また、ポートの持つ属性はバインドされるリソースによって決まるが、ポートの属性とバインドされるリソースは以下のように分類される。

【0037】A. トランザクション・ポート (transaction port)

トランザクションを行なうポートであり、1つのポート

で送受信が可能とされ、リソースとしてノード内のアドレスがバインドされる。

B. アイソクロナス・トーカー (isochronous talker)
アイソクロナス・データの送信を行なうポートであり、リソースとしてアイソクロナス・チャンネル番号とアイソクロナス帯域がバインドされる。

C. アイソクロナス・リスナー (isochronous listener)

アイソクロナス・データの受信を行なうポートであり、リソースとしてアイソクロナス・チャンネル番号がバインドされ、そのチャンネル番号で送出されたパケットを受信することができる。

D. マルチキャスト・トーカー (multicast talker)

マルチキャスト・データの送信を行なうポートであり、リソースとしてマルチキャスト・チャンネル番号がバインドされる。

E. マルチキャスト・リスナー (multicast listener)

マルチキャスト・データの受信を行なうポートであり、リソースとしてマルチキャスト・チャンネル番号がバインドされ、そのチャンネル番号で送出されたパケットを受信することができる。

【0038】なお、mLANTランスポート層10でのトランザクションはトランザクション層3のレベルで見れば(ポートにバインドされた)アドレスに対するトランザクションである。この方法によれば、mLANTランスポート層10で提供されているサービスを利用してポートにバインドされていないアドレス(例えば Configuration ROM 内のエントリ)に対するアクセスは行なえないことになる。そこで、これを行なうために各ノード上に内部情報を提供するためのNIM(ノード情報管理)9が定義されている。他のノードはこのNIM9に対して要求を行なうことによって、ノード情報(Node Unique ID、ノード内のポートの数、各ポートの属性など)を獲得することができる。

【0039】2. 2 mLANTランスポート層の構成
mLANTランスポート層10は前記したように以下に挙げる5つのモジュールで構成されている。

A. アイソクロナス・マネージャ7

アイソクロナス転送を行なうためのリソース(アイソクロナス・チャンネル番号と帯域)を獲得し、アイソクロナス・トーカーにバインドする。また、各ノードで獲得されたアイソクロナス・リソースと、そのリソースのバインド情報を管理する。

【0040】B. マルチキャスト・マネージャ6

マルチキャスト転送を行なうためのチャンネルを獲得し、マルチキャスト・トーカーにバインドする。また、各ノードで獲得されたマルチキャスト・チャンネルと、そのチャンネルのバインド情報を管理する。

C. トランザクション・マネージャ5

トランザクション・ポートに対するリソースの割り当て

を管理する。

D. PIM (パス情報管理) 8

アイソクロナス・ポート／マルチキャスト・ポートにパスを設定する機能を提供し、設定されたパスの管理を行なう。

E. NIM (ノード情報管理) 9

ノード・コントローラ4の上位モジュールとして機能する。ノード・コントローラ4から通知されたパスの状態変化にしたがって他のモジュールに対して設定変更などの操作を行なう。

【0041】2. 3 アイソクロナス転送サービス

次に、アイソクロナス転送サービスについての説明を行う。アイソクロナス転送を行なう送信ポートはアイソクロナス・トーカーと呼ばれ、受信ポートはアイソクロナス・リスナーと呼ばれる。アイソクロナス転送では、例えば、 $125\mu\text{sec}$ の周期でサイクルが設定されており、アイソクロナス・チャンネル番号を確保しているノードはこのサイクルの間に1回だけ確保した帯域分のデータを転送することができる。

【0042】新規に作成されたポートをアイソクロナス・トーカーとしてアイソクロナス転送方式でのデータ送信を行う場合は、そのポートに対してアイソクロナス・チャンネル番号と帯域をバインドする必要がある。アイソクロナス転送方式でのデータ転送は特定のノードに対してパケットが転送されるのではなく、アイソクロナス・チャンネル番号を付されたパケットが全ノード宛にブロードキャストされる。アイソクロナス転送をサポートしているそれぞれのノードは、受信した全てのアイソクロナス・パケットをリンク層2に通知する。リンク層2はパケットに付されたチャンネル番号を見て転送されたデータを取り込むかどうかを決定する。

【0043】2. 3. 1 アイソクロナス・マネージャ
アイソクロナス転送において、アイソクロナス・マネージャ7は各ノード上にあるアイソクロナス・リソースを管理するmLANトランスポート層10内部のモジュールであり、アイソクロナス・マネージャ7は上位アプリケーションからの要求によってアイソクロナス・リソース(アイソクロナス・チャンネル番号とアイソクロナス帯域)を獲得し、それらをアイソクロナス・トーカーにバインドする。

【0044】このバインドの手続きによってアイソクロナス・トーカーは送信を行なうことができるようになる。また、アイソクロナス・リスナーに対してはアイソクロナス・チャンネル番号がバインドされる。チャンネル番号をバインドされたアイソクロナス・リスナーはそのチャンネル番号で送信されるアイソクロナス・データを受信することができる。また、アイソクロナス・マネージャ7は以上の手続きによって得たリソースを保持する。そしてバスリセットが発生してバスの初期化が行なわれた後、アイソクロナス・ポートに対して保持してい

たリソースの再バインドを行なう。

【0045】2. 3. 2 アイソクロナス・リソースの確保とバインド処理

次に、アイソクロナス・トーカーに対するアイソクロナス・リソースの確保とバインド処理について、図10に示すフローチャートを参照しながら説明する。アイソクロナス・マネージャ7がPIM8からのアイソクロナス・リソース・バインド要求(Isochronous Manager Bind Resource Request)を受け取ると、アイソクロナス・リソースの確保とバインド処理が開始され、以下の手順でアイソクロナス・リソースが獲得され、そのリソースがアイソクロナス・トーカーに対してバインドされる。まず、ステップS10にてアイソクロナス・マネージャ7はバス・マネージャに対してアイソクロナス・チャンネル番号と帯域の割り当て要求を行なう。これはバス・マネージャ上のCSRに対して1対1の排他的なトランザクションにより行なわれる。

【0046】次いで、上位層からの要求通りにアイソクロナス・リソースを獲得できたか否かがステップS20にて判定されるが、獲得できたと判定された場合、ステップS30にてアイソクロナス・マネージャ7はアイソクロナス・トーカーに対してそれらのリソースをバインドする。ただし、実際には後述するトーカー情報変更処理においてバインド相当の処理が行なわれ、このためこの処理は破線で示されている。次いで、ステップS40にてアイソクロナス・トーカーが作成されたことを示すために、獲得したリソースの内容を添付した図9に示すフォーマットとされたトーカー情報報告パケットをブロードキャストする。そして、ステップS50にてPIM8にリソース・バインドの成否を通知するが、この場合はリソース・バインドが成功しているのでその旨を通知する。

【0047】なお、トーカー情報報告パケットは、図9に示すように、トーカー情報報告パケットであることを示すIDに続いて、トーカーのポートID、トーカーのノードID、トーカーのノードユニークID、トーカーの獲得したリソース情報とから構成される。また、ステップS20にてアイソクロナス・リソースが獲得できなかったと判定された場合は、ステップS50にてリソースバインドの不成功の通知をPIM8に対して行う。以上の処理が終了すると、リターンされて、アイソクロナス・トーカーはアイソクロナス転送を行なうことができるようになる。

【0048】2. 3. 3 アイソクロナス・リソース・バインド処理(リスナー)

ところで、アイソクロナス転送の受信を行なう場合、アイソクロナス・マネージャ7はアイソクロナス・リスナーに対してリソースをバインドする必要がある。このアイソクロナス・リソース・バインド処理を図11に示すフローチャートを参照しながら説明する。アイソクロナ

10

20

30

40

50

ス・マネージャ7が上位層からのアイソクロナス・リソース・バインド要求を受け取ると、アイソクロナス・リソースの確保とバインド処理が開始され、以下の手順でアイソクロナス・リソースがアイソクロナス・リスナーに対してバインドされる。

【0049】まず、リスナー側のアイソクロナス・マネージャ7はステップS60にてリスナーと指定されたポートに、指定されたトーカーのアイソクロナス・チャンネル番号をバインドする。この場合は、チャンネル番号や帯域を新たに確保する必要はなく、単に受信チャンネルの番号をリスナーにバインドするだけである。ただし、リスナーにバインドされるアイソクロナス・チャンネル番号はすでにトーカーによって確保されているチャンネルの番号でなければならない。そして、ステップS70にてアイソクロナス情報テーブルに新たにエントリされたリスナー・ポートに関する情報が書き込まれる。このアイソクロナス情報テーブルを図7に示すが、このテーブルには、ポートID、ポートタイプ、アイソクロナス・リソースが登録されている。

【0050】次いで、ステップS80にてアイソクロナス・マネージャ7がアイソクロナス情報テーブルによって保持している受信チャンネルリストがリンク層2に渡される。リンク層2は、すべてのアイソクロナス・パケットを受信しているが、この受信チャンネルリストにあるチャンネル番号のアイソクロナス・パケットを受信した場合のみ、アイソクロナス・マネージャ7に受信通知を行なう。次いで、ステップS90にてPIM8にリソース・バインドの成否を通知するが、通常はリソース・バインドに失敗することはないので、通常はリソース・バインド成功の通知がPIM8に行なわれる。以上により、リスナーにおけるアイソクロナス・リソース・バインド処理が終了する。

【0051】2. 3. 4 アイソクロナス送信データ受付処理、アイソクロナス送信処理

このようにして、アイソクロナス・トーカーおよびアイソクロナス・リスナーにアイソクロナス・リソースがバインドされて、アイソクロナス転送を行えるようになる。アイソクロナス転送を行なう場合は、まずアイソクロナス送信データの受付処理が行なわれて、次にアイソクロナス送信処理が行なわれるようになる。そこで、アイソクロナス送信データ受付処理を図12(a)に示すフローチャートを、アイソクロナス送信処理を同図(b)に示すフローチャートを参照しながら説明する。

【0052】PIM8からアイソクロナス・マネージャ7が、アイソクロナス・トーカーから転送されるアイソクロナス送信データの転送要求を受け取ると、アイソクロナス送信データ受付処理が開始され、ステップS100にてアイソクロナス・マネージャ7は、まず内部に保持している図7に示すアイソクロナス情報テーブルを参照して、指定されたアイソクロナス・トーカーにバイン

ドされているアイソクロナス・チャンネル番号を獲得する。次いで、ステップS110にて転送すべき転送データとアイソクロナス・チャンネル番号とを一時保存する。これは、アイソクロナス転送が所定サイクルの開始通知を受けて行なわれるので、サイクルが開始されるまで待機するためである。

【0053】アイソクロナス送信データ受付処理が終了し、リンク層2よりアイソクロナス・サイクルの開始が通知されると、アイソクロナス送信処理が開始される。この処理は、まず、ステップS120にてアイソクロナス・マネージャ7が獲得されたアイソクロナス・チャンネル番号を使用して、リンク層2に対してアイソクロナス転送要求を発行する。この場合、転送データもリンク層2に渡されて、アイソクロナス転送データは、リンク層2を介して物理層1から全ノードにブロードキャストされるようになる。

【0054】2. 3. 5 アイソクロナス受信処理
このようにしてブロードキャストされたアイソクロナス転送されたパケットの受信を行なう場合のアイソクロナス受信処理を、図13(a)に示すフローチャートを参照して説明する。アイソクロナス・マネージャ7がリンク層2からの受信通知を受け取ると、アイソクロナス受信処理が開始され、ステップS130にてアイソクロナス・マネージャ7は内部の図7に示すアイソクロナス情報テーブルを参照して、アイソクロナス・チャンネル番号からアイソクロナス・パケットを渡すリスナーのポート番号を取得する。そして、ステップS140にて、取得したポートに受信通知を行ない、受信データをそのポートを指定して転送する。

【0055】2. 3. 6 トーカー重複検出処理
ところで、1つのアイソクロナス・チャンネル番号がバインドされるアイソクロナス・トーカーはただ1つである。すなわち、特定のアイソクロナス・チャンネル番号でパケットを送出しているアイソクロナス・トーカーはただ1つとされる。したがって、複数のアイソクロナス・トーカーから同一のアイソクロナス・チャンネル番号でパケットが送信されていることを、リンク層2からの通知によりアイソクロナス・マネージャ7が検出した場合、上位層に対する受信通知をリンク層2において中止し、アイソクロナス・チャンネル番号が重複していることを上位層に通知する。

【0056】この場合に行なわれるトーカー重複検出処理のフローチャートを図13(b)に示すが、トーカーの重複が検出された場合、ステップS150にてトーカーの重複していることが、リンク層2よりの重複受信の通知を受けて、アイソクロナス・マネージャ7はPIM8へ通知する。この場合、重複しているトーカーに関係のあるポート番号が指定されてPIM8に通知される。また、PIM8に対するアイソクロナス転送の受信通知はリンク層2のレベルで中止される。

【0057】2. 3. 7 下位層とのコミュニケーション

以上説明したアイソクロナス転送サービスにおいて、mLANトランスポート層10のアイソクロナス・マネージャ7はリンク層2と、次のようなサービス・プリミティブによりコミュニケーションを行なう。mLANトランスポート層10は、Link Isochronous Request のサービス・プリミティブを利用してリンク層2に対してアイソクロナス・パケットの転送を要求する。また、Link Isochronous Indication のサービス・プリミティブは、リンク層2からアイソクロナス・マネージャ7に対して発行され、アイソクロナス・パケットを受信したことが通知される。

【0058】さらに、Link Cycle Start Indication のサービス・プリミティブはリンク層2からアイソクロナス・マネージャ7に対して発行され、アイソクロナス・サイクルが開始されたことを示す。アイソクロナス・マネージャ7はこの通知を受け取ったあと、送信すべきデータがあった場合に Link Isochronous Request のサービス・プリミティブによってデータの転送を行なう。さらに、アイソクロナス・マネージャ7はLink Isochronous Control Requestのサービス・プリミティブを利用して、ノード内のアイソクロナス・リスナーにバインドされている全てのアイソクロナス・チャンネル番号を Channel Receive List としてリンク層2に渡す。リンク層2はこのリストに含まれているチャンネルを持つアイソクロナス・パケットを受信した場合にアイソクロナス・マネージャ7に対して Link Isochronous Indication を発行する。

【0059】2. 3. 8 PIMとのコミュニケーション

次に、アイソクロナス・マネージャ7とPIM8とのコミュニケーションについて説明する。Isochronous Manager Data Request のサービス・プリミティブは、PIM8からアイソクロナス・マネージャ7に対して発行され、アイソクロナス・マネージャ7はこれを受けて指定されたデータの転送を行なう。Isochronous Manager Data Indication のサービス・プリミティブは、アイソクロナス・マネージャ7からPIM8に対して発行され、アイソクロナス・パケットを受信したことがPIM8に対して通知される。Isochronous Manager Bind Resource Request のサービス・プリミティブは、PIM8からアイソクロナス・マネージャ7に対して発行され、アイソクロナス・マネージャ7は指定されたポートに対してリソースのバインドを行なう。

【0060】Isochronous Manager Bind Resource Confirmation のサービス・プリミティブは、アイソクロナス・マネージャ7からPIM8に対して発行され、Isochronous Manager Bind Resource Request によるバインド要求に対する成否が返される。Isochronous Manager

Release Resource Request 、および、Isochronous Manager Release Resource Confirmation のサービス・プリミティブは、獲得したリソースをリリースするためのものである。

【0061】2. 3. 9 バスリセット対応処理、バスリセット完了通知受信処理

ところで、新たなノードがバスに参入された場合は、バスリセットを行なってすべてのバスをリセットし、次いでノードにノードIDを新たに付け替えてバスを再構築する。この場合、バスリセット対応処理に続いてバスリセット完了通知受信処理が行なわれる。バスリセット対応処理を図15に示すフローチャートを参照して説明すると、バスリセットが発生するとNIM9からアイソクロナス・マネージャ7に対してリセット要求 (Isochronous Manager control request (Reset))が発行され、バスリセット対応処理が開始される。このリセット要求を受け取ったアイソクロナス・マネージャ7は、ステップS190にてPIM8からのアイソクロナス転送要求をrejectする。そして、バスリセット対応処理は終了する。この場合、アイソクロナス情報テーブルで保持されているアイソクロナス・トーカーとアイソクロナス・チャンネル番号は保持される。

【0062】そして、バスリセットが完了するとNIM9からアイソクロナス・マネージャ7に対してイニシャライズ要求 (Isochronous Manager control request (Initialize))が発行される。このイニシャライズ要求を受け取ったアイソクロナス・マネージャ7は、バスリセット完了通知受信処理を行なう。バスリセット完了通知受信処理を図16に示すフローチャートを参照しながら説明すると、まず、アイソクロナス・マネージャ7は、ステップS200にて図7に示すアイソクロナス情報テーブルから1つのTalker entryを取り出して、ステップS210にてバス・マネージャにアイソクロナス・チャンネル番号と帯域の割当を要求する。これはバス・マネージャ上のCSRに対して1対1の排他的なトランザクションにより行なわれる。

【0063】次いで、要求通りにアイソクロナス・リソースを獲得できたか否かがステップS220にて判定されるが、獲得できたと判定された場合、ステップS230にてアイソクロナス・マネージャ7はアイソクロナス・トーカーに対してそれらのリソースをバインドする。ただし、実際には前記したトーカー情報変更処理においてバインド相当の処理が行なわれ、このためこの処理は破線で示されている。次いで、ステップS240にてアイソクロナス・トーカーが作成されたことを示すために、獲得したリソースの内容を添付した図9に示すフォーマットとされたトーカー情報報告パケットをブロードキャストする。そして、ステップS250にてPIM8にリソースバインドの成否を通知するが、この場合はリソースバインドが成功しているのでその旨が通知され

10

20

30

40

50

る。

【0064】また、要求通りにアイソクロナス・リソースを獲得できなかったとステップS220にて判定された場合は、ステップS250にてリソースバインドの不成功がPIM8に通知される。このステップS250の処理が終了すると、ステップS260にてアイソクロナス・リソースをバインドしていない Talker entry が残っているか否かが判定されて、残っていると判定された場合は、ステップ200ないしステップS250のアイソクロナス・トーカーにリソースをバインドする処理が、再度行なわれることにより、すべての Talker entry にアイソクロナス・リソースをバインドする処理が行なわれ、この処理は終了する。これによって、アイソクロナス・マネージャ7は外部からの要求を受け付けることができるようになる。

【0065】2. 3. 10 トーカー情報変更処理
次に、トーカー情報報告パケットをNIM9が受信したことがアイソクロナス・マネージャ7に通知 (Isochronous Manager control request (Talker Info Received)) されると、talker 情報変更処理が行なわれる。このtalker 情報変更処理の説明を図14に示すフローチャートを参照しながら次に行なう。まず、ステップS160にて受信された図9に示すフォーマットのトーカー情報報告パケットから、トーカー・ポートID、トーカー・リソース情報を取り出して、新たなトーカーの entry を作成して、図7に示すアイソクロナス情報テーブルの更新を行なう。次いで、ステップS170にてトーカー情報報告パケットをPIM8に転送することにより、PIM8にパス情報の変更を報告する。この場合、PIM8は後述するように、パス情報テーブルを書き換えてパスの接続処理を行なう。

【0066】さらに、ステップS180にてNIM9にアイソクロナス情報テーブルの更新結果を報告して、この処理は終了する。この talker 情報変更処理は、ノードにトーカーが作成された際にブロードキャストされるトーカー情報報告パケットの受信により行なわれるが、他のノードからブロードキャストされたトーカー情報報告パケットに限らず、自ノードからブロードキャストされたトーカー情報報告パケットによっても行なわれる処理である。

【0067】2. 3. 11 NIMとのコミュニケーション

なお、NIM9からアイソクロナス・マネージャ7に発行されるIsochronous Manager Control Request のサービス・プリミティブにより以下のパラメータが渡される。

・RESET : アイソクロナス・マネージャ7は内部状態のリセットを行なう。

・INITIALIZE : アイソクロナス・マネージャ7は内部状態の初期化を行なう。

・TALKER INFO RECEIVED : NIM9がトーカー情報報告パケットを受信したことを通知する。

また、アイソクロナス・マネージャ7からNIM9に対して発行されるIsochronous Manager Control Confirmation のサービス・プリミティブにより、Isochronous Manager Control Requestによる制御要求に対する成否が返される。さらに、アイソクロナス・マネージャ7からNIM9に対して発行される Isochronous Manager Event Indication のサービス・プリミティブにより、アイソクロナス・マネージャ7の内部状態の変化を報告する。

【0068】2. 4 マルチキャスト転送サービス

次に、マルチキャスト転送サービスについての説明を行なう。マルチキャスト転送は1つの送信ポートから複数の受信ポートに対して行なわれる非同期コネクションレスのデータ転送方式である。マルチキャスト転送方式でパケットを送出するポートがマルチキャスト・トーカー、このパケットを受信するポートがマルチキャスト・リスナーであり、マルチキャスト転送はmLANトランスポート層10で定義されている。アプリケーション・ユーザはマルチキャスト・トーカーでの転送を行なう前にポートに対してマルチキャスト・チャンネル番号をバインドしておく必要がある。これは、そのポートがそのチャンネルにおける送信権を獲得することを意味している。

【0069】あるマルチキャスト・チャンネル番号を使ってパケットを送信できるのはそのチャンネル番号がバインドされているマルチキャスト・トーカーだけであり、マルチキャスト・トーカーから送信されるパケットは特定のマルチキャスト・リスナーに送られるのではなく、トランザクション層3で定義されているブロードキャスト機能によって全ノードに対して転送される。各ノードがマルチキャスト・パケットを受信した場合、トランザクション層3からマルチキャスト・マネージャ6に対して受信通知が発行される。受信されたパケットにはマルチキャスト・トーカーに割り当てられてたマルチキャスト・チャンネル番号が添えられているので、受信側のマルチキャスト・マネージャ6はこのチャンネル番号を検査して実際に受信すべきパケットであるかどうかを判断する。したがって、マルチキャスト・トーカーは送信したパケットをどのマルチキャスト・リスナーが受信しているのかわからない。

【0070】マルチキャスト・トーカーにバインドされるマルチキャスト・チャンネル番号は、バインドを行なう際にmLANトランスポート層10内部のマルチキャスト・マネージャ6が確保する。また、マルチキャスト・リスナーにバインドされるマルチキャスト・チャンネル番号は、すでにいずれかのマルチキャスト・トーカーにバインドされている番号とされる。

【0071】マルチキャスト転送の転送メカニズムを図

3を参照しながら説明すると、この図ではノード#1のポート#1とポート#3がマルチキャスト・トーカーとして機能している。それぞれのポートにはマルチキャスト・チャンネル番号の#2と#4がバインドされている。例えばノード#1のポート#1に対して上位アプリケーションからデータ転送要求が発行されると、そのデータはマルチキャスト・チャンネル#2でローカルバス中の全ノードにパケットとしてブロードキャストされる。各ノードは、送出されたパケットをブロードキャスト機能により受信するため、受信されたパケットは下位層からmLANトランスポート層10に渡されて、そのパケットを受信する。

【0072】一方、ノード#2ではポート#2が、ノード#3ではポート#3がマルチキャスト・チャンネル#2を受信するためのマルチキャスト・リスナーとして機能しているので、それぞれのノードのmLANトランスポート層10は、マルチキャスト・チャンネル#2で受信したパケット内のデータをそれぞれのマルチキャスト・リスナーに渡す。これにより、1対多のマルチキャスト転送が実現される。同様にノード#1のポート#3から送出されたパケットはマルチキャスト・チャンネル#4によって各ノードに転送され、ノード#2ではポート#3（マルチキャスト・チャンネル#4がバインドされたマルチキャスト・リスナー）によって受信される。ノード#3ではマルチキャスト・チャンネル#4はどのポートにもバインドされていないのでそのチャンネルでmLANトランスポート層10が受信したパケットは破棄される。

【0073】2. 4. 1 マルチキャスト・マネージャ
マルチキャスト・マネージャ (Multicast Manager) 6は、マルチキャスト転送を行なうポートのためにマルチキャスト・チャンネル番号を獲得し、それをポートにバインドする機能を持つモジュールである。マルチキャスト・マネージャ6はトーカー情報報告パケットを発行することによってマルチキャスト・チャンネルを獲得する。また、ローカルバス内で発行されるトーカー情報報告パケットを全て受信し、それぞれのマルチキャスト・トーカーとバインドされているマルチキャスト・チャンネル番号の対応テーブルを、図6に示すマルチキャスト情報テーブルとして保持する。

【0074】トーカー情報報告パケットを受信したマルチキャスト・マネージャ6は受信順によってそのトーカーのマルチキャスト・チャンネル番号を決定し、図6に示すマルチキャスト情報テーブルを作成する。この場合、自ノードから送出されたトーカー情報報告パケットを受信したマルチキャスト・マネージャ6は、自ノードのマルチキャスト・チャンネル番号を決定することになる。また、この対応テーブルはマルチキャスト・リスナーがマルチキャスト・トーカーにバインドされているマルチキャスト・チャンネル番号を探すために使われる。

マルチキャスト・マネージャ6は上位アプリケーションからの要求によってその情報を提供する。マルチキャスト転送を行なうポート（マルチキャスト・トーカー / マルチキャスト・リスナー）を持つノードにはマルチキャスト・マネージャ6が実装される。

【0075】2. 4. 2 マルチキャスト・リソースの確保とバインド処理

次に、マルチキャスト・リソースの確保とバインド処理について、図17に示すフローチャートを参照しながら説明する。マルチキャスト・トーカーとなるポートにバインドされるマルチキャスト・リソースの確保とバインド処理は、マルチキャスト・マネージャ6が行なう。上位層であるPIM8からマルチキャスト・チャンネル・バインド要求 (MulticastManager Bind Resource Request) を受け取ると開始され、ステップS600にてマルチキャスト・マネージャ6は、獲得したマルチキャスト・リソースであるマルチキャスト・チャンネル番号を、Node Unique ID とポートIDをパラメータとして持つ図9に示すフォーマットのトーカー情報報告パケットに添付して全ノードにブロードキャストする。

【0076】そして、ステップS610にてマルチキャスト・チャンネル番号が確保されたと判断された場合には、確保されたリソース（マルチキャスト・チャンネル番号）を指定されたマルチキャスト・トーカーにバインドする。次いで、PIM8にリソースバインドの成否の通知を行なうが、この場合には成功の通知が行なわれる。また、マルチキャスト・チャンネル番号が確保できなかったと判断された場合には、ステップS610からステップS630に進み、この旨の通知がPIM8に行なわれる。これで、この処理は終了する。なお、ステップS610およびステップS620の処理が破線で示されているのは、実際にはトーカー情報報告パケットをブロードキャストすることにより、後述するトーカー情報変更処理において、リソース・バインドに相当する処理が行なわれるためである。

【0077】2. 4. 3 トーカー情報変更処理

ところで、ブロードキャストされたトーカー情報報告パケットは各ノードのNIM9により受信されて、トーカー情報変更処理が行なわれる。このトーカー情報変更処理を、図21に示すフローチャートを参照しながら説明する。トーカー情報変更処理は、各ノード上のNIM9からトーカー情報報告パケットの受信通知がマルチキャスト・マネージャ6に対して行なわれると開始され、まずステップS750にてマルチキャスト・チャンネル番号を、トーカー情報報告パケットの受信順に各ノードにおいて割り当てるために、各ノードのマルチキャスト・マネージャ6に備えられたパケット・カウンタを”1”だけインクリメントする。すなわち、マルチキャスト・マネージャ6はバスリセットによって”0”にリセットされるパケット・カウンタを持ち、バスリセット後に送

信された全てのトーカ-情報報告パケットを受信してその数をカウントしている。

【0078】次に、ステップS760にてマルチキャスト・マネージャ6は受信したトーカ-情報報告パケットに基づいて Node Unique ID、ポートID、マルチキャスト・チャンネル番号の対応テーブルである図6に示すマルチキャスト情報テーブルを更新する。この場合、トーカ-情報報告パケットからポートIDが取り出されると共に、その時のパケット・カウンタ値がリソース情報として利用されて、Talker の entry が作成される。次いで、ステップS770にてPIM8にトーカ-情報報告パケットを転送してパス情報の更新を報告し、さらにステップS780にてNIM9に更新結果を報告してこの処理は終了する。すなわち、トーカ-情報報告パケットをブロードキャストしたマルチキャスト・マネージャ6は、自分自身が送信したパケットを受信した時点でのカウンタの値を、自ノードのマルチキャスト・トーカ-にバインドするマルチキャスト・チャンネル番号として使用することになる。このマルチキャスト・チャンネル番号はローカルバス内で最大256個（0～255）まで割り当てることができる。なお、動作中にカウンタが256を越えてしまった場合、マルチキャスト・マネージャ8は上位層に対して通知を行なう。

【0079】2. 4. 4 マルチキャスト送信処理

次に、マルチキャスト送信処理を図19に示すフローチャートを参照しながら説明すると、マルチキャスト送信処理は、上位層であるPIM8からマルチキャスト・マネージャ6が転送データを受信した時に開始される。すると、ステップS670にて転送データを受信したマルチキャスト・マネージャ6は、まず内部に保持している図6に示すマルチキャスト情報テーブルを参照して、送出すべきマルチキャスト・トーカ-にバインドされているマルチキャスト・チャンネル番号を獲得する。次いで、ステップS680にて獲得されたマルチキャスト・チャンネル番号を指定し、転送データをトランザクション層3に送ることにより、マルチキャスト転送要求を発行する。これにより、転送データはマルチキャスト転送されるが、この場合、マルチキャスト転送は実際にはブロードキャストによる送信であるため ACK は返らない。

【0080】2. 4. 5 マルチキャスト・リソースのバインド処理（リスナー）

次に、マルチキャスト・リスナーについて説明を行なうが、マルチキャスト・リスナーにおいてマルチキャスト・パケットを受信するためにはそのマルチキャスト・パケットに含まれているマルチキャスト・チャンネル番号がリスナーにバインドされている必要がある。このために行なわれるマルチキャスト・リソースのバインド処理を図18に示すフローチャートを参照しながら説明すると、この処理はPIM8よりリソースバインドの要求

（Multicast Manager Bind Resource Request）を受信すると開始される。まず、ステップS640にて内部に保持されているマルチキャスト情報テーブルが参照されて、受信すべきマルチキャスト・チャンネル番号が、指定されたポートにバインドされる。

【0081】次いで、ステップS650にてマルチキャスト情報テーブルにそのポートID、ポートタイプ、マルチキャスト・リソース（マルチキャスト・チャンネル番号）が新たな entry として書き込まれ、さらに、ステップS660にてPIM8にリソースバインドの成否が通知される。ただし、リソース・バインドにはトーカ-で既に確保されているチャンネルにバインドするような要求しかないので、リソース・バインドに失敗することはない。したがって、ここでは成功の通知がPIM8に行なわれて、この処理は終了する。このようにして、マルチキャスト・リスナーにマルチキャスト・リソースがバインドされると、マルチキャスト転送を受信できるようになる。

【0082】2. 4. 6 マルチキャスト受信処理

次に、マルチキャスト転送されたマルチキャスト・パケットを受信するマルチキャスト受信処理を図20に示すフローチャートを参照しながら説明する。この処理は、トランザクション層3からの受信通知により開始され、まず、ステップS700にてトーカ-が重複しているか否かが検出される。これは、マルチキャスト・パケットに含まれるマルチキャスト・チャンネル番号とは、そのパケットを送出したマルチキャスト・トーカ-にバインドされているマルチキャスト・チャンネル番号であり、1つのマルチキャスト・チャンネル番号がバインドされるマルチキャスト・トーカ-はただ1つである。すなわち、特定のマルチキャスト・チャンネル番号でパケットを送出しているマルチキャスト・トーカ-はただ1つとなるからである。

【0083】そこで、複数のマルチキャスト・トーカ-から同一のマルチキャスト・チャンネル番号でパケットが送信されていることをマルチキャスト・マネージャ6が検出した場合、ステップS730に進み上位層であるPIM8に対する受信通知を中止すると共に、重複しているマルチキャスト・チャンネル番号に関連するポート番号を指定して重複していることをPIM8に通知する。また、ステップS700にてトーカ-が重複されていないと検出された場合は、ステップS710に進みマルチキャスト・マネージャ6は、マルチキャスト情報テーブルを参照してリスナーのポートを取得する。次いで、ステップS720にて各マルチキャスト・リスナーにバインドされているチャンネル番号と一致した場合のみ、そのポートに対してパケット受信の通知を行なう。

【0084】なお、下位層であるトランザクション層3から受信通知により、全てのマルチキャスト・パケットをマルチキャスト・マネージャ6は受信するが、パケッ

ト内に示されるマルチキャスト・チャンネル番号が、自ノードの各マルチキャスト・リスナーにバインドされている番号と一致した場合のみ、そのポートに対してパケット受信の通知を行なう。これにより、リスナーはマルチキャスト・データを受信することができる。また、該当するリスナーが複数ある場合にはマルチキャスト・マネージャ6がデータの複製を行う。そして、マルチキャスト・チャンネル番号がどのリスナーにもバインドされていないパケットは破棄される。以上でマルチキャスト受信処理は終了する。

【0085】2. 4. 7 下位層とのコミュニケーション

以上説明したようにmLANトランスポート層10のマルチキャスト・マネージャ6は、マルチキャスト転送を行なうためにトランザクション層3とコミュニケーションを行なう。この時、mLANトランスポート層10はトランザクション層3で定義される以下のサービス・プリミティブを利用する。

Transaction Data Request: マルチキャスト・マネージャ6はマルチキャスト転送を開始するためにトランザクション層3に対してこのプリミティブを発行する。マルチキャスト転送はトランザクション層3のブロードキャスト機能を利用して行なわれるので、Transaction TypeとしてBROADCAST WRITEが指定される。

Transaction Data Indication: トランザクション層3からマルチキャスト・マネージャ6に対してブロードキャスト・データの受信が通知される。

【0086】2. 4. 8 PIMとのコミュニケーション

また、マルチキャスト・マネージャ6は、マルチキャスト転送を行なうためにPIM8ともコミュニケーションを行なう。この時、マルチキャスト・マネージャ6は上位モジュールであるPIM8とコミュニケーションするために以下のようなサービス・プリミティブが定義される。

Multicast Manager Data Request: PIM8からマルチキャスト・マネージャ6に対して発行される。マルチキャスト・マネージャ6は指定されたデータの転送を行なう。

Multicast Manager Data Indication: マルチキャスト・マネージャ6からPIM8に対して発行される。マルチキャスト・パケットを受信したことをPIM8に対して通知する。

【0087】Multicast Manager Bind Resource Request: PIM8からマルチキャスト・マネージャ6に対して発行される。マルチキャスト・マネージャ6は指定されたポートに対してリソースのバインドを行なう。このサービス・プリミティブによって、ポートIDおよびマルチキャスト・チャンネル番号(ポートIDによって指定されるポートのポートタイプがMULTICAST LISTENERである場合に必要。)のパラメータが渡される。

Multicast Manager Bind Resource Confirmation: マルチキャスト・マネージャ6からPIM8に対して発行され、Multicast Manager Bind Resource requestによるバインド要求に対する成否が返される。

【0088】2. 4. 9 NIMとのコミュニケーション

また、マルチキャスト・マネージャ6は、マルチキャスト転送を行なうためにNIM9ともコミュニケーションを行なう。この時、マルチキャスト・マネージャ6は上位モジュールであるNIM9とコミュニケーションするために以下のようなサービス・プリミティブが定義され、NIM9はマルチキャスト・マネージャ6の内部のリセットや初期化を行なうためにこれらのサービス・プリミティブを利用する。

【0089】Multicast Manager Control Request: NIM9からマルチキャスト・マネージャ6に対して発行される。このサービス・プリミティブによって以下のパラメータが渡される。

・RESET: マルチキャスト・マネージャ6は内部状態のリセットを行なう。

・INITIALIZE: マルチキャスト・マネージャ6は内部状態の初期化を行なう。

・TALKER INFO RECEIVED: NIM9がトーカー情報報告パケットを受信したことを通知する。

Multicast Manager Control Confirmation: マルチキャスト・マネージャ6からNIM9に対して発行される。Multicast Manager CONTROL.requestによる制御要求に対する成否が返される。

Multicast Manager Event Indication: マルチキャスト・マネージャ6からNIM6に対して発行される。マルチキャスト・マネージャの内部状態の変化を報告する。

【0090】2. 4. 10 バスリセット対応処理、バスリセット完了通知受信処理

次に、バスリセット時の動作について説明するが、前記したように新たなノードがバスに参入された場合は、バスリセットを行なってすべてのバスをリセットし、次いで各ノードにノードIDを新たに付け替えてバスを再構築する。この場合、バスリセット対応処理に続いてバスリセット完了通知受信処理が行なわれる。バスリセット対応処理を図22に示すフローチャートを参照しながら説明すると、バスリセットが発生するとNIM9からマルチキャスト・マネージャ6に対してリセット要求(Multicast Manager control request (Reset))が発行され、バスリセット対応処理が開始される。すると、ステップS800にてこのリセット要求を受け取ったマルチキャスト・マネージャ6は、トーカー情報報告パケットカウンタを"0"にリセットする。さらに、マルチキャスト情報テーブルで保持されているマルチキャスト・チャンネル番号をクリアすると共に、ステップS810にてPIM8からのマルチキャスト転送要求をrejectす

る。そして、バスリセット対応処理は終了する。

【0091】そして、バスリセットが完了するとNIM9からマルチキャスト・マネージャ6に対してインシャライズ要求(Multicast Manager control request (Initialize))が発行される。このインシャライズ要求を受け取ったマルチキャスト・マネージャ6は、バスリセット完了通知受信処理を行なう。バスリセット完了通知受信処理を図23に示すフローチャートを参照しながら説明すると、内部のマルチキャスト情報テーブルでポートIDが保持されており、かつそのポートIDに対応するマルチキャスト・チャンネル番号がクリアされていた場合、そのポートにバインドされていたマルチキャスト・チャンネル番号がバスリセットによってクリアされたことを意味している。

【0092】そこで、ステップS820にてマルチキャスト情報テーブルからリソースがクリアされた1つのTalker entryを取り出し、ステップS830にてマルチキャスト・マネージャ6は、そのTalker entryに対するマルチキャスト・チャンネル番号を獲得し、リソース(マルチキャスト・チャンネル番号)の内容を添付したトーカー情報報告パケットをブロードキャストする。そして、ステップS840にてリソースが確保されたか否かが判断されるが、確保されたと判断された場合はステップS850に進み、取り出されたTalker entryに対するマルチキャスト・チャンネル番号がバインドされる。次いで、ステップS860にてPIM8にリソースバインドの成否を通知するが、この場合はリソース・バインドが成功しているのでその旨が通知される。

【0093】また、ステップS840にてリソースの確保に失敗したと判断された場合は、リソース・バインドの不成功がPIM8に通知される。なお、ステップS840およびステップS850の処理が破線とされているのは、前記2.4.5 マルチキャスト・リソースの確保とバインド処理で述べた理由と同じ理由による。この処理が終了すると、ステップS870にてまだマルチキャスト・リソースがバインドされていないTalker entryが残っているか否か判定されて、残っていると判定された場合は、ステップS820に戻り前記マルチキャスト・トーカーにリソースをバインドする処理が、再度行なわれることにより、すべてのTalker entryにマルチキャスト・リソースをバインドする処理が行なわれ、Talker entryのすべてにリソースがバインドされると、この処理は終了する。これによって、マルチキャスト・マネージャ6は外部からの要求を受け付けることができるようになる。

【0094】2.5 PIM (バス情報管理)

次に、mLANトランスポート層10内における上位層であるPIM (バス情報管理) 8の詳細な説明する。PIM8はmLANトランスポート層10の中に存在し、トーカー・ポートとリスナー・ポートの間にバスを設定

するモジュールである。PIM8は設定したバスに関する情報を保持し、バスリセットによってバスの再初期化が行なわれた場合にも設定されていたバスを自動的に再設定する機能を有している。ただし、PIM8が保持するバス情報はmLANトランスポート層10で定義されている前記した3つの転送方式のうち、アイソクロナス転送とマルチキャスト転送を行なうポートに関してのみである。

【0095】すなわち、PIM8は上位層に対して以下のようなサービスを提供する。

A. バスの設定 / 解放: アイソクロナス転送やマルチキャスト転送を行なうポート間にバスを設定する。また、設定されているバスを解放することができる。

B. バス情報の記憶: PIM8が提供するサービスを利用して設定されたバスを記憶する。PIMは電源を落とした状態でもこのバス情報を保持し、このバス情報を使って電源再投入後のバスの再構成を行なう。

C. 電源再投入後のバスの再構成: 電源再投入によるバスの初期化後、PIM8は保持しているバス情報をもとに自動的にバスの再設定を行ない、バス初期化前の接続情報を再現する。

D. 他ノード上のPIM8とのバス情報の交換: PIM8はバス情報管理プロトコルしたがって上位アプリケーションの要求によって他ノード上のバス情報を問い合わせる。この場合の他ノードとは、例えばフロッピーディスクドライブである。

【0096】さらに、PIM8はローカル・ノード上にあるリスナー・ポートとリモート・ノードのトーカー・ポートの間でバスを設定/削除するサービスを提供している。したがって、PIM8はリモート・ノードにあるトーカーとリスナーの間にバスを設定することはできない。この機能は上位アプリケーションであるバス・マネージャによって実現される。

【0097】2.5.1 バスの設定処理

次に、PIM8が行なうバスの設定処理を図24に示すフローチャートを参照しながら説明する。バスの設定処理は、リスナー・ノード上のPIM8が、上位アプリケーションからのバス設定要求(PIM Set Path Request)を受け取ると、開始される。まず、上位アプリケーションからバス設定要求を受け取ったPIM8は、ステップS300にてトーカー・ポートに関する属性を獲得する。次に、ステップS310にてリスナーポートの属性を、獲得されたトーカー・ポートと同じ属性に合わせる。

【0098】そして、ステップS320にてPIM8は、下位のチャンネル管理モジュール(ポートタイプがアイソクロナスの場合はアイソクロナス・マネージャ7、ポートタイプがマルチキャストの場合はマルチキャスト・マネージャ6)にバインドを要求する。この場合、トーカーについてはすでにバインドが完了している

10

20

30

40

50

ので、このバインドはリスナーについてのみ行なわれる。次いで、バスが正常に設定されたとして、ステップS330にてこのバス情報によりバス情報テーブルを変更する。この場合の変更は、接続状態フラグが Disconnected であったものを Connected に書き換える変更となる。

【0099】そして、ステップS340にて上位層に対してバス設定要求に対する成否を報告するが、この場合は成功の報告が行なわれる。なお、このバスの設定はトーカーで既に確保されているリソースを、リスナーのリソースにコピーする処理であるので通常は失敗することはない。これにより、バスの設定処理は終了する。なお、バス設定処理においては1つのトーカーポートに対して複数のリスナー・ポートへ接続するバスを設定することはできるが、逆に1つのリスナー・ポートを複数のトーカー・ポートに接続するバスを設定することはできない。また、バスの設定はリスナー上のPIM8によって行なわれ、PIM8のレベルではトーカーがリスナーを指定してバスを設定することはできない。

【0100】2. 5. 2 バスに対するデータ送信処理
次に、バスに対するデータ送信処理について、図25に示すフローチャートを参照しながら説明する。上位層よりPIM8が送信するデータを受信すると、バスに対するデータ送信処理が開始され、まず、ステップS360にてNIM9が保持する図8に示すポート情報エントリが参照されることにより、データを送信するポートタイプが参照される。そして、ステップS370にてポートタイプがトーカーか否かが判定され、ポートタイプがトーカーと判定されると、さらにステップS380にてポートタイプがマルチキャストか否かが判定される。ここで、ポートがマルチキャストのポートと判定されると、ステップS390にてPIM8はマルチキャスト・マネージャ6にデータを転送する。マルチキャスト・マネージャ6では、これを受けて前記したマルチキャスト送信処理が行なわれる。

【0101】また、ポートがアイソクロナスポートと判定された場合は、ステップS380からステップS400に進み、アイソクロナス・マネージャ7にデータを転送する。アイソクロナス・マネージャ7では、これを受けて、前記図12に示すアイソクロナス送信処理が行なわれる。以上でバスに対するデータ送信処理は終了する。なお、ステップS370にてポートがトーカーと判定されない場合は、そのまま終了する。

【0102】2. 5. 3 バスの解放処理

次に、バスの解放処理を図26に示すフローチャートを参照しながら説明するが、すでに設定されているバス情報は、リスナー・ノード上のPIM8が保持するバス情報テーブルに記憶されているので、実際にバスを解放する作業はリスナー・ノードのPIM8で行われる。この場合、トーカーに対する確認を行なうことなくバスの解

放が行なわれる。PIM8は上位層からバス解放要求(PIM Release Path Request)を受け取ると、バスの解放処理が開始される。まず、ステップS410にてPIM8は、図4に示すバス情報テーブルを参照してリスナー・ポートに対して該当するようなバスが接続されているかどうかを確認する。

【0103】そして、ステップS420にてバスが接続されているか否かの判断を行ない、バスが接続されていた場合にはステップS430にてリスナー・ポートにバインドされているリソースを解放し、バス情報テーブルから対応するエントリを削除する。次いで、ステップS440にて上位層に対してバス開放要求の返答を行なうが、この場合はバス開放の返答を行なう。また、ステップS420にてバスが接続されているか否かの判断を行ない、バスが接続されていないと判断された場合は、上位層に対してその旨の返答をステップS440にて行なう。

【0104】2. 5. 4 バスリセット対応処理、トーカー情報変更処理

各種不成功通知処理

ところで、mLANトランスポート層10では、非同期マルチキャスト転送に用いるチャンネルをダイナミックに獲得しているため、バスリセットが発生した場合、トーカーは新たにマルチキャスト・チャンネル番号を獲得する必要がある。したがって、バスリセットの前後ではトーカーにバインドされるマルチキャスト・チャンネル番号が変更される可能性がある。したがって、リスナーには新たにトーカーにバインドされたマルチキャスト・チャンネル番号を設定しないと、データを受信することができなくなる。

【0105】そこでPIM8は、バスリセット時にバスリセット対応処理をまず行ない、次にトーカー情報変更処理を行なう。バスリセット対応処理を図27を参照しながら説明する。バスリセットが発生するとNIM9からPIM8に対してリセット要求(PIMcontrol request (Reset))が発行されると、バスリセット対応処理が開始される。PIM8はこの要求を受け取ると、ステップS450にてバス情報テーブル内の全てのエントリの接続状態フラグを DISCONNECT に設定する。つまりバス情報テーブルは保持しているが、このバス情報に基づく転送は行なえない状態とする。

【0106】また、前記したようにバスリセットが完了すると各ノードはそれぞれのノード内のアイソクロナス・トーカーとマルチキャスト・トーカーに対してリソースの再バインドを行なう。ここで正常にリソースの再バインドが行なわれた場合、このトーカーについての更新された情報が、図9に示すフォーマットのトーカー情報報告パケットによってバス内の全ノードにブロードキャストされる。そして、このパケットの受信はマルチキャスト・マネージャ6かアイソクロナス・マネージャ7を

通してPIM8に対して通知され(IM event indication()あるいはMM event indication())、トーカー情報変更処理が開始される。

【0107】このトーカー情報変更処理のフローチャートを図29に示すが、まずステップS460にて各ノード上のPIM8はバス情報テーブルを参照して、ノード・ユニーク・IDを検出することにより、通知が行なわれたトーカーに対応するエントリが探される。次いで、そのトーカー・ポートの属性がステップS470にて獲得され、さらに、そのトーカー・ポートとバスによって接続されるリスナー・ポートの属性がステップS480にて獲得される。

【0108】そして、ステップS490にて獲得された両ポートを接続することが可能か否かが判断され、可能と判断された場合は、さらにステップS500にてポートのタイプがマルチキャストか否かが判断される。そして、マルチキャストと判断された場合はステップS510にてマルチキャスト・マネージャ6にマルチキャスト・リソースのバインドを要求し、アイソクロナスと判定された場合はステップS520にてアイソクロナス・マネージャ7にアイソクロナス・リソースのバインドを要求する。これを受けたいずれかの下位のチャンネル管理モジュールは、該当するリスナーにリソースのバインド処理を行ない、その結果をPIM8に通知する。この通知を受けてPIM8は、ステップS530にてリソースのバインドが成功したか否か判断し、バインドが成功したと判断した場合はステップS540にてバス情報テーブルの接続状態フラグをConnectedに変更する。

【0109】なお、ステップS490にて獲得された両ポートを接続することが不可能と判断された場合は、ステップS550にてエラー処理が行なわれる。また、ステップS530にて下位のチャンネル管理モジュールからバインドが不成功と通知された場合も、ステップS550にてエラー処理が行なわれる。以上の処理が行なわれてトーカー情報変更処理は終了する。PIM8はトーカー情報報告パケットを受け取るたびに、このトーカー情報変更処理を行なう。しかし、例えばバスリセット以前に転送を行っていたトーカーのノードが引き抜かれたような場合には、バスリセット完了後にそのトーカーはバス上に存在しないので、このトーカーに関するトーカー情報報告パケットはブロードキャストされないことになる。このような場合、このトーカーに関するバス情報エントリはDISCONNECTEDの状態では保持されるので、情報としては保持されているがバスは再設定されないことになる。

【0110】このような場合、バス情報としてはそのまま保持されているので、次のバスリセットでまたバスに接続されトーカー情報報告パケットがブロードキャストされればバスの再接続を行なうことができる。また、PIM8はこのような「保持されているが接続されてい

い」バス情報を消去するためのサービスを上位層に対して提供することもできる。この作業によって上位アプリケーションはバスリセットの前後でデータ転送を継続して行なうことができる。また、PIM8には各種不成功通知が下位モジュールから通知されるが、この場合には図28に示す各種不成功通知処理が行なわれる。この処理では、PIM8はステップS560にて上位層に問題発生を通知して終了する。

【0111】2. 5. 5 NIMとのコミュニケーション

PIM8はNIM9とコミュニケーションを行なうために以下のようなサービス・プリミティブが定義されている。

PIM Control Request

PIM Control Confirmation

PIM Event Indication

NIM9はPIM8の内部のリセットや初期化を行なうためにこれらのサービス・プリミティブを利用する。

【0112】2. 6 NIM(ノード情報管理)9

NIM9は各ノードのmLANトランスポート層10内に存在し、以下のような機能を有している。

A. バスの状態変化にもとづくmLANトランスポート層10の初期化

NIM9はノード・コントローラ4の上位層として位置し、ノード・コントローラ4から通知されるバス状態の変化にしたがって、mLANトランスポート層10内部の各モジュールのリセット/初期化を行なう。

【0113】B. 他ノードに対するノード情報の提供
mLANトランスポート層10ではポート間でデータを交換するためのサービスが提供されているが、上位アプリケーションは他ノードの内部アドレスに対して直接アクセスすることができない。各ノード上の情報や機能設定のためのレジスタが内部のアドレス上に配置されているので、上位アプリケーションはこれらのアドレスに対して操作を行なう場合にはNIM9を通じて行なう。NIM9は他ノードからの問い合わせを受け付けるためのトランザクション・ポートを1つ持つ。このポートをNIMポートと呼ぶ。NIMポートのポートIDは全ノード共通であり、全てのノードはそのIDをあらかじめ知っている必要がある。NIM9はConfiguration ROMのエントリやポート情報エントリを参照して問い合わせ要求のあった情報をrequesterに返す。

【0114】C. トーカー情報報告パケットの処理
トーカー情報報告パケットは、前記したようにアイソクロナス・トーカーやマルチキャスト・トーカーが作成された時にそれぞれのマネージャによって発行されるパケットである。このパケットはトーカー・ポートに関する情報を報告するためにローカルバス内の全ノードのNIMポートに対して発行される。NIM9はこのパケットを受信するとパケットの内容をノード内のアイソクロナ

ス・マネージャ7かマルチキャスト・マネージャ6に渡す。どちらのマネージャに渡すかはトーク・ポートにバインドされているリソースによって決定する。

【0115】次に以上の機能を奏するためのNIM9の動作について説明する。

2. 6. 1 ポート情報の問い合わせ処理

上位アプリケーションがノード内のリスナー・ポートにリソースをバインドしようとする場合、このアプリケーションは特定のトーク・ポートにバインドされているリソースを知らなければならない。このような場合、上位アプリケーションはNIM9に問い合わせしてトーク・ポートに関する情報を得るようにする。このポート情報の問い合わせ処理を、図30に示すフローチャートを参照しながら説明する。この処理は、上位層からNIM9は問い合わせ要求を受け取ると開始され、ステップS900にてポート情報テーブルを参照することにより、問い合わせのあったentryの情報を入手して上位層に返答する。また、他ノードのポート情報については、そのノード上のNIM9とトランザクション通信を行なうことにより、トーク・ポートに関するポートタイプおよびリソース情報を入手し、その情報を上位層に対して通知する。これによりこの処理は終了する。

【0116】2. 6. 2 トーカー情報報告パケット処理

NIM9は、トーカー情報報告パケットを受信すると、トーカー情報報告パケット処理を行なうが、この処理を図31に示すフローチャートを参照しながら説明する。トーカー情報報告パケットをNIM9が受信すると、トーカー情報報告パケット処理が開始され、まずステップS910にて受信したトーカー情報報告パケットに応じて、ノード情報テーブルにノードIDとノード・ユニークIDを書き込むことにより更新する。次に、ステップS920にてトーカー情報報告パケット内のトークにバインドされているリソース情報から、トークがアイソクロナス・トークかマルチキャスト・トークかを判断する。そして、アイソクロナス・トークと判断された場合は、ステップS930に進みアイソクロナスマネージャ7にトーカー情報報告パケットを転送し、マルチキャスト・トークと判断された場合は、ステップS940にてマルチキャストマネージャ6にトーカー情報報告パケットを転送する。これで、この処理は終了する。

【0117】2. 6. 3 下位層とのコミュニケーション

NIM9はノード・コントローラ4によって定義されている以下のサービス・プリミティブを利用してノード・コントローラ4とコミュニケーションを行なう。

SB Control Request (SB_CONTROL.request)

SB Control Confirmation (SB_CONTROL.confirmation)

SB Event indication (SB_EVENT.indication)

【0118】2. 6. 4 バスリセット対応処理、バスリセット完了時処理

バスリセットが発生するとノード・コントローラ4からNIM9に対してバスリセット要求 (SB event indication (BUS RESET START)) が発行され、この通知をNIM9が受け取ると、バスリセット対応処理が開始される。この処理を図32に示すフローチャートを参照しながら説明する。バスリセット要求がNIM9で受信されると、ステップS950にてNIM9は、mLANトランスポート層10内部のアイソクロナス・マネージャ7に対して内部状態のクリアを要求するためのリセット要求 (Isochronous Manager Control Request (Reset)) を発行する。次いで、ステップS960にてマルチキャスト・マネージャ6に対して内部状態のクリアを要求するためのリセット要求 (Multicast Manager Control Request (Reset)) を発行する。さらに、ステップS970にてPIM8に対して内部状態のクリアを要求するためのリセット要求 (PIM Control Request (Reset)) を発行し、この処理は終了する。

【0119】また、バスリセットが完了するとノード・コントローラ4からNIM9に対してバスリセット完了 (SB event indication (BUS RESET COMPLETE)) が発行される。NIM9が、この通知を受け取ると図33に示すフローチャートのバスリセット完了時処理が開始される。バスリセット完了時処理が開始されると、ステップS980にてNIM9はmLANトランスポート層10内部のアイソクロナス・マネージャ7にリセット完了を発行し、次いで、ステップS990にてマルチキャスト・マネージャ6にリセット完了を発行する。これで、この処理は終了する。

【0120】2. 7 mLANトランスポート層のファシリティ

2. 7. 1 パス情報テーブル

パス情報テーブルはノード上のリスナー・ポート (アイソクロナス・リスナー/ マルチキャスト・リスナー) に設定されているパスを記憶しておくためのテーブルである。このパス情報テーブルはPIM8によって管理されている。また、このテーブルの内容はNIM9からも参照されNIM9によって他ノードに渡される。パス情報テーブルのフォーマットを図4に示すが、パス情報テーブルで1つのパスに関する情報がパス情報エントリである。1つのパス情報エントリには以下の情報が含まれる。

【0121】A. リスナーのポートID (16ビット) : ローカルノードのポート番号。

B. トーカーのポートID (16ビット) : リモートノードのポート番号。

C. トーカーがあるノードの Node Unique ID (64ビット) : リモートノードを特定するために用いるIDであり、64ビットの長さを持ち、各メーカによって各ノ

ドごとにあらかじめ設定されている。バスリセットやパワーリセットによって変更されることがない。

D. 接続状態フラグ：このエントリで示されるバスの接続状態を示すフラグ。バス情報テーブル内に記憶されていても再初期化後のネットワーク構成の変化によってバスの再設定が行なわれないエントリもある（例えばリモートノードが見つからなかった場合）。このような場合にこのエントリのバスが実際に設定されているかどうかを示すフラグである。

【0122】2. 7. 2 ノード情報テーブル
ノード情報テーブルはローカルバス内のノードのノードIDと Node Unique ID の対応を示すテーブルである。ノード情報テーブルはNIM9によって管理される。mLANではバスリセットによってノードIDが変更される場合があるので、バスリセット後はNIM9によって更新が行なわれる。ノード情報テーブルは次の情報からなり、そのフォーマットを図5に示す。

A. ノードID：バスの初期化時に各ノードに動的に割り当てられるID。

B. Node Unique ID：各ノードにあらかじめ割り当てられている64ビットのIDである。mLANでは上位24ビットを Node Vendor ID とし、下位40ビットを Vendor Unique ID とする。これによって Node Unique ID の一意性が保証されている。

【0123】2. 7. 3 マルチキャスト情報テーブル
マルチキャスト・マネージャ6によって管理される情報テーブル。ノード内のマルチキャスト・ポートのポートIDと、そのポートにバインドされているマルチキャスト・リソースの対応を示すテーブルである。それぞれのマルチキャスト・ポートに対する情報テーブルのエントリには、次の情報が含まれ、そのフォーマットを図6に示す。

A. ポートID：マルチキャスト・リソースがバインドされているポートのIDである。

B. ポートタイプ：マルチキャスト・トーカーあるいはマルチキャスト・リスナーのいずれかを示している。

C. マルチキャスト・リソース：マルチキャスト・チャンネル番号である。

【0124】なお、バスリセットが発生するとマルチキャスト・リソース（チャンネル番号）はクリアされる。マルチキャスト・ポートのポートIDに関する内容はバスリセットの前で保持されるので、バスリセット完了後それぞれのポートに対して新たなマルチキャスト・チャンネル番号を割り当てるための処理が行なわれる。ただし、このテーブルの内容はパワーリセット（電源のON/OFF）では保存されない。

【0125】2. 7. 4 アイソクロナス情報テーブル
アイソクロナス・マネージャ7によって管理される情報テーブル。ノード内のアイソクロナス・ポートのポートIDと、そのポートにバインドされているアイソクロナ

ス・リソースの対応を示すテーブルであり、それぞれのアイソクロナス・ポートに対する情報テーブルのエントリには、以下の情報が含まれ、そのフォーマットを図7に示す。

【0126】A. ポートID：アイソクロナス・リソースがバインドされているポートのIDである。

B. ポートタイプ：アイソクロナス・トーカーあるいはアイソクロナス・リスナーのいずれかを示す。

C. アイソクロナス・リソース：バインドされているリソースを示す。アイソクロナス・トーカーに対してはアイソクロナス・チャンネル番号と帯域、アイソクロナス・リスナーに対してはアイソクロナス・チャンネル番号がバインドされている。なお、アイソクロナス情報テーブルの内容はバスリセットが発生した場合にも保持されている。

【0127】2. 7. 5 ポート情報エントリ
ポート情報エントリはNIM9によって管理され、ノード内に作成されるポートの属性が記述される。ポート情報エントリは各ノード内のアドレス空間上に配置される。各ポートの情報エントリはポートIDから一意に指定されるオフセット上に配置される。リモート・ノードから直接ポート情報エントリのアドレスにアクセスすることはできない。それぞれのポート情報エントリにはポートタイプに応じたリソースに関するエントリが含まれ、そのフォーマットを図8に示す。

【0128】A. ポートID：ポートを特定するために各ポートに割り当てられるID（16bit）であり、ポートIDは1つのノード内での一意性が保証されている。特定の用途に使われるポート（例、PIM8/NIM9など）のポートIDは固定的に割り当てられ、全ノードで等しいIDが割り当てられるようにする。

B. ポートタイプ：ポートが扱うことのできる転送方式を示す。4ビット（d d d d）で転送方式の種類（マルチキャスト / アイソクロナス / トランザクション）を示し、別の4ビット（k k k k）で転送方向（送信 / 受信 / 送受信）を示す。

C. リソース：ポートにバインドされているリソースを示す。ポートタイプに応じたリソースとなる。

【0129】2. 7. 6 トーカー情報報告パケット
このパケットはトーカーが作成されて、リソースがバインドされたことをローカルバス中の全ノードに報告するためのパケットである。このパケットは各ノードのNIM9に対してブロードキャストされる。マルチキャスト・トーカーの場合にはポートが作成された時に、このパケットがブロードキャストされる。各ノードのトランスポート層10はこのパケットの受信順でマルチキャスト・トーカーに対して割り当てるマルチキャスト・チャンネル番号を決定する。アイソクロナス・トーカーの場合には各ノードのアイソクロナス・マネージャがリソースを確保した時点で発行される。トーカー情報報告パケッ

トには以下の情報が含まれ、そのフォーマットを図9に示す。

【0130】A. パケットID: トーカー情報パケットを示すパケットIDである。

B. トーカー・ポートID: 作成されたトーカーのポートIDである。

C. トーカー・ノードID: トーカーが属するノードのノードIDである。

D. トーカー・ユニークID: 前記したとおり、各ノードにあらかじめ割り当てられている64ビットのIDである。

E. トーカー・リソース情報: 作成されたトーカーにバインドされているリソース情報である。

【0131】2. 8 mL ANサイクル・ストラクチャー

以上説明したmLANのサイクル・ストラクチャーの一例を図34に示す。この図では、サイクル#mのサイクルが主に示されており、この例では各1転送サイクル

(nominal cycle period)は125μsecとされている。この転送サイクルの開始タイミング(cycle sync)はサイクル・マスターから全ノードに通知される。アイソクロナス・マネージャ7はこれを受信して送信すべきデータがある時にはアイソクロナス転送を行なう。この転送は、物理層1のアービトレーションにより1ノードだけが転送できるように制御される。この結果、例えば図示するようにアイソクロナス・チャンネルJ、チャンネルK、・・・チャンネルNでそれぞれ確保した帯域分のデータがそれぞれのチャンネルで順次転送される。なお、アイソクロナス転送においては各転送サイクルにおいて帯域が確保されているので、各転送サイクルごとにデータは転送される。

【0132】アイソクロナス転送が終了すると、マルチキャスト・マネージャ6によるマルチキャスト転送が行なわれる。この転送も1ノードだけが転送できるように制御される。この例では、チャンネルA、チャンネルB、チャンネルCがマルチキャスト・チャンネルとされている。なお、このマルチキャスト転送においては帯域はある幅の中で任意とされるためパケットが1転送サイクル期間中で終了しない場合が生じる。この場合は、1転送サイクルを越えてパケットが終了するまでマルチキャスト転送が継続される。そして、図示するように次のサイクル(#m+1)の開始が遅延される。マルチキャスト転送は、送出するデータがある時にだけ転送すれば良いため、離散的に発生するデータを送出するには好適な転送方式である。

【0133】2. 9 マルチキャスト転送とアイソクロナス転送との使い分け

アプリケーションより見た場合、ノード間の情報は、あるポートを指定して情報を送出/受信するだけで、マルチキャスト転送でもアイソクロナス転送でもまったく同

じように転送されているように見える。したがって、同じMIDIメッセージでもデータの帯域や、アイソクロナス・チャンネルの空き具合に応じて、マルチキャスト転送で転送したり、アイソクロナス転送で転送したりすることができる。

【0134】この場合、トーカーに対応するポート・タイプはアプリケーションの実行の状態に応じて変化するものとなるが、リスナーのポート・タイプはトーカーのポート・タイプと一致させる必要があるので、PIM8の行なうパスの設定処理において、トーカーのポート・タイプをリスナーに引き継ぐ(コピーする)ようにする。これにより、アプリケーションのユーザ(特にリスナー)はアイソクロナス転送か、マルチキャスト転送かを考慮することなく、パスを設定することができるようになる。ここで、トーカー側のアプリケーションでは、利用者が転送の方式を設定するようにしてもよく、アイソクロナス・チャンネルの帯域の利用状況を見て、アプリケーションが自動的に判別するようにしてもよい。

【0135】2. 10 プラグ・アンド・プレイ
プラグ・アンド・プレイは各種電子機器をケーブル等に接続するだけで、煩雑な内部設定をすることなく電子機器を利用可能にする技術である。mLANにおいては、すべての情報をバス情報が必要なマルチキャスト転送またはアイソクロナス転送で転送するので、パスの設定を行っていない機器(ノード)同士を接続した場合、何も信号の送受を行なうことができず、機器として動作できないこととなる。そこで、本発明のmLANにおいてプラグ・アンド・プレイを実現するために、Well-knownチャンネルを定義する。初期状態においては、特定のプロトコルを特定のチャンネル番号で送受する。これにより、取りあえずデータの送受をプロトコルごとに行なうことができる。

【0136】Well-knownチャンネルは、バス情報がまったく設定されていない場合における動作を保証するものであるが、バス情報が設定されているネットワークにおいて新たな機器(ノード)が参入された場合にも、何もデータを送受できないことが起こりえる。受信の場合は、ノード追加時に発生されるバスリセットに応じて、トーカー情報報告パケットがブロードキャストされるので、それを元にトーカーとバインドされているチャンネル番号を検出し、対応するプロトコルのトーカーに対応するリスナー・ポートを複数作成することにより、受信することができる。この場合、上位層のアプリケーションは、それらの複数のポートからの情報をミックスして利用することになる。送信の場合は、リスナーの状態に影響を与えることなく、情報を受信させることが不可能であるので、この場合は何も行なわない。

【0137】なお、MIDIメッセージはそのままパケット化して転送することが効率の上からは良い。また、リアルタイム性が要求とされず、信頼性が要求されるよ

うなデータについては、例えば通常のファイル転送のような別プロトコルとして転送しても良い。また、デジタル・オーディオデータはストリームとしてアイソクロナス転送方式で転送するのが好適である。

【0138】

【発明の効果】本発明は以上のように構成されているので、物理的な接続形態に依存しない仮想的な経路（論理的なパス）をネットワークの内部に構築することができる。すなわち、そのネットワークに接続されていれば、それぞれの機器の位置関係に関係なく2つの機器間に経路を設定し、その経路を使ってデータを交換することができる。これによって接続形態に関わらず仮想的な経路を確立することができるので、例えば機器の繋ぎ順を間違えることなくデータを転送できないというミスがなくなる。また、論理的に接続されているため、物理的な接続を変更せずに機器間の接続を変更することができるようになる。

【0139】仮想的な経路のパス情報はネットワークに接続されている各機器がそれぞれ記憶する。そして、ネットワーク内の全てのパス情報を一括して別メディアに保存すれば、再構成を行なう場合でもそれをネットワークにロードすることによって素早く正確に構成を再現することができる。さらに、本発明はこのようなネットワークにおいて、所定の帯域を確保してデータを伝送することのできるアイソクロナス転送に加えて、転送データがある時に指定された複数のノードにデータの転送を行えるマルチキャスト転送を行うこともできる。したがって、離散的なデータか否かにかかわらず、効率的な伝送を可能とすることができる。すなわち、リアルタイム性を要求されるMIDIデータやオーディオデータ等を効率よく伝送することができるようになる。

【図面の簡単な説明】

【図1】 本発明のmLANの通信形態の概要を示す図である。

【図2】 本発明のmLANのプロトコル階層を示す図である。

【図3】 マルチキャスト転送を説明するための図である。

【図4】 パス情報テーブルのフォーマットを示す図である。

【図5】 ノード情報テーブルのフォーマットを示す図である。

【図6】 マルチキャスト情報テーブルのフォーマットを示す図である。

【図7】 アイソクロナス情報テーブルのフォーマットを示す図である。

【図8】 ポート情報エントリーのフォーマットを示す図である。

【図9】 トーカー情報報告パケットのフォーマットを示す図である。

【図10】 アイソクロナス・マネージャにおけるアイソクロナス・リソースの確保とバインド処理のフローチャートを示す図である。

【図11】 アイソクロナス・マネージャにおけるアイソクロナス・リソース・バインド処理のフローチャートを示す図である。

【図12】 アイソクロナス・マネージャにおけるアイソクロナス送信データ受付処理およびアイソクロナス送信処理のフローチャートを示す図である。

10 【図13】 アイソクロナス・マネージャにおけるアイソクロナス受信処理およびトーカー重複検出処理のフローチャートを示す図である。

【図14】 アイソクロナス・マネージャにおけるトーカー情報変更処理のフローチャートを示す図である。

【図15】 アイソクロナス・マネージャにおけるバスリセット対応処理のフローチャートを示す図である。

【図16】 アイソクロナス・マネージャにおけるバスリセット完了通知受信処理のフローチャートを示す図である。

20 【図17】 マルチキャスト・マネージャにおけるマルチキャスト・リソースの確保とバインド処理のフローチャートを示す図である。

【図18】 マルチキャスト・マネージャにおけるマルチキャスト・リソース・バインド処理のフローチャートを示す図である。

【図19】 マルチキャスト・マネージャにおけるマルチキャスト送信処理のフローチャートを示す図である。

【図20】 マルチキャスト・マネージャにおけるマルチキャスト受信処理のフローチャートを示す図である。

30 【図21】 マルチキャスト・マネージャにおけるトーカー情報変更処理のフローチャートを示す図である。

【図22】 マルチキャスト・マネージャにおけるバスリセット対応処理のフローチャートを示す図である。

【図23】 マルチキャスト・マネージャにおけるバスリセット完了通知受信処理のフローチャートを示す図である。

【図24】 PIMにおけるバスの設定処理のフローチャートを示す図である。

40 【図25】 PIMにおけるバスに対するデータ送信処理のフローチャートを示す図である。

【図26】 PIMにおけるバスの解放処理のフローチャートを示す図である。

【図27】 PIMにおけるバスリセット対応処理のフローチャートを示す図である。

【図28】 PIMにおける各種不成功通知処理のフローチャートを示す図である。

【図29】 PIMにおけるトーカー情報変更処理のフローチャートを示す図である。

50 【図30】 NIMにおけるポート情報問い合わせ処理のフローチャートを示す図である。

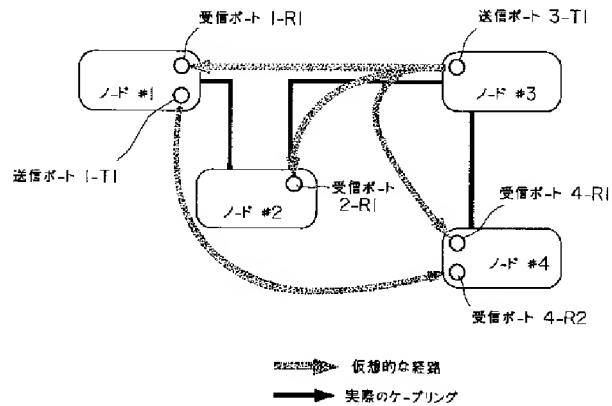
【図31】 NIMにおけるトークー情報報告パケット処理のフローチャートを示す図である。

【図32】 NIMにおけるバスリセット対応処理のフローチャートを示す図である。

【図33】 NIMにおけるバスリセット完了時処理のフローチャートを示す図である。

【図34】 本発明のmLANのサイクルストラクチャーの一例を示す図である。 *

【図1】

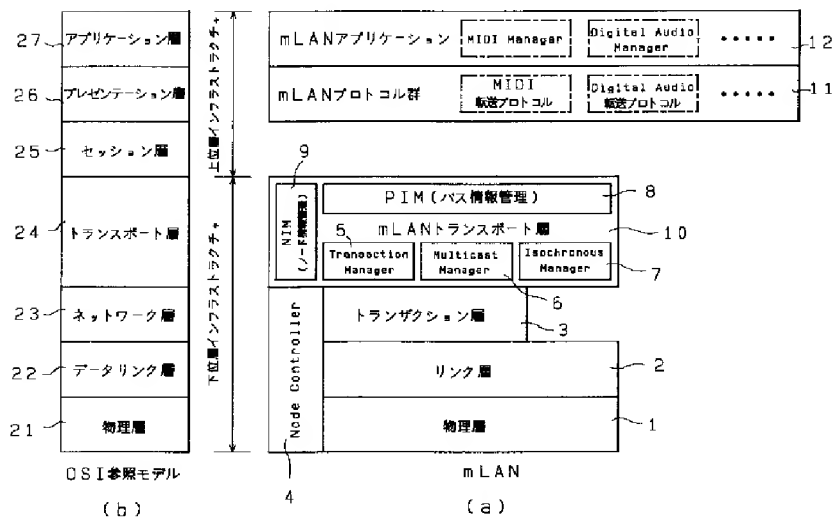


【図5】

ノード情報テーブル (NIM)

ノードID	Node Unique ID
ID_No.	Unique ID No.

【図2】



【図4】

バス情報テーブル (PIM)

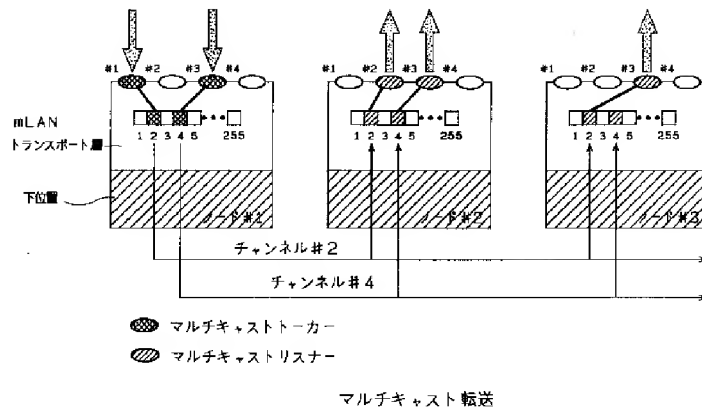
Listener ポートID	Talker ポートID	Talker Node Unique ID	接続状態フラグ
ID_No.	ID_No.	Unique ID No.	Connected/Disconnected

【図6】

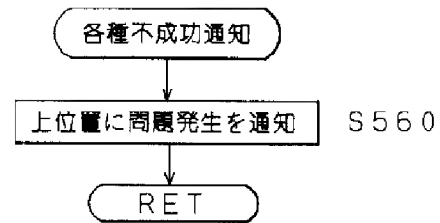
MC情報テーブル (MC Manager)

ポートID	ポートタイプ	MCリソース
ID_No.	MC Talker	チャンネル番号
ID_No.	MC Listener	チャンネル番号

【図3】



【図28】



【図7】

ISO情報テーブル (ISO Manager)

ポートID	ポートタイプ	ISOリソース
ID_No.	ISO Talker	チャンネル番号/帯域
ID_No.	ISO Listener	チャンネル番号

【図8】

ポート情報エントリー (NIM)

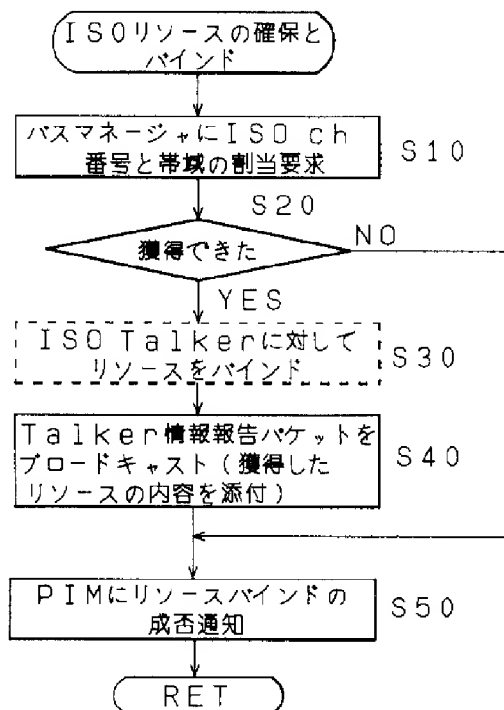
ポートID	ポートタイプ	リソース
ID_No.	ISO Talker	チャンネル番号/帯域
ID_No.	ISO Listener	チャンネル番号
ID_No.	MC Listener	チャンネル番号

【図9】

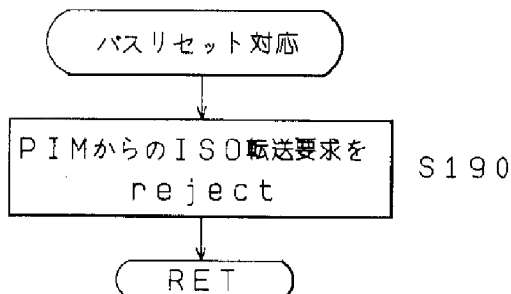
Talker情報報告パケット

Packet ID	Talker ポートID	Talker ノードID	Talker Node Unique ID	Talker リソース情報
-----------	-----------------	-----------------	--------------------------	------------------

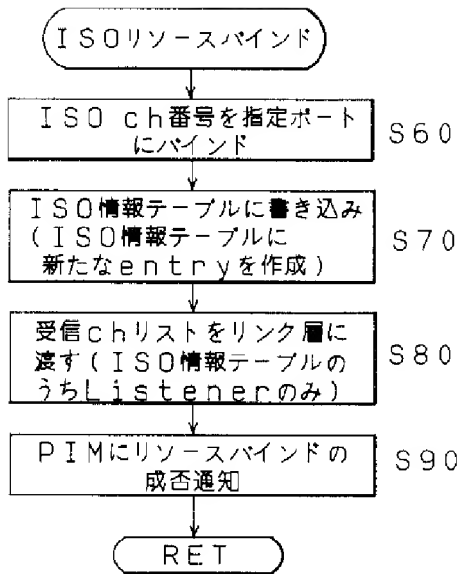
【図10】



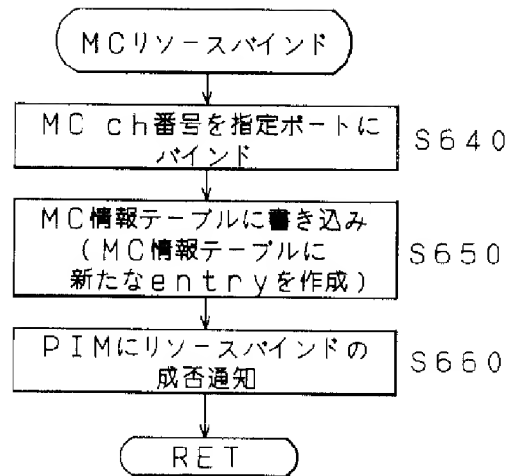
【図15】



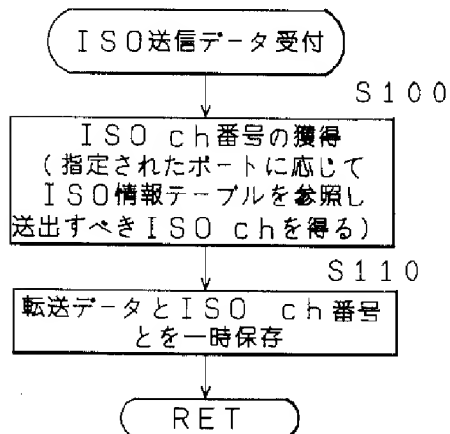
【図11】



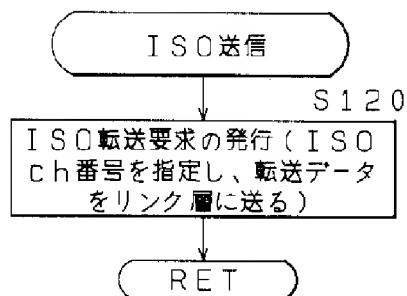
【図18】



【図12】

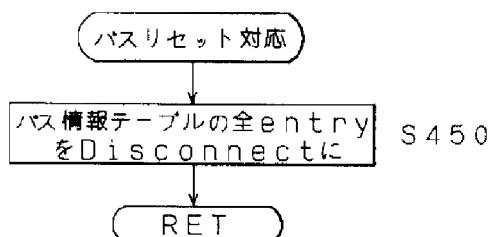


(a)

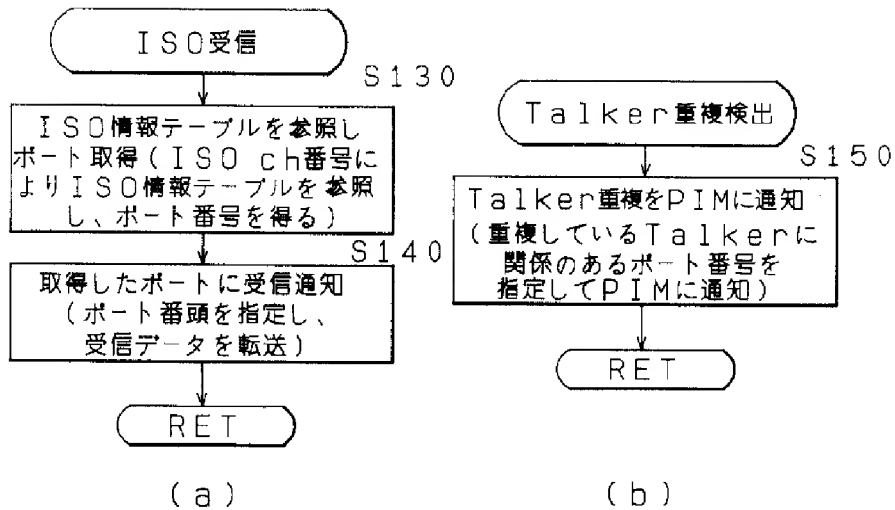


(b)

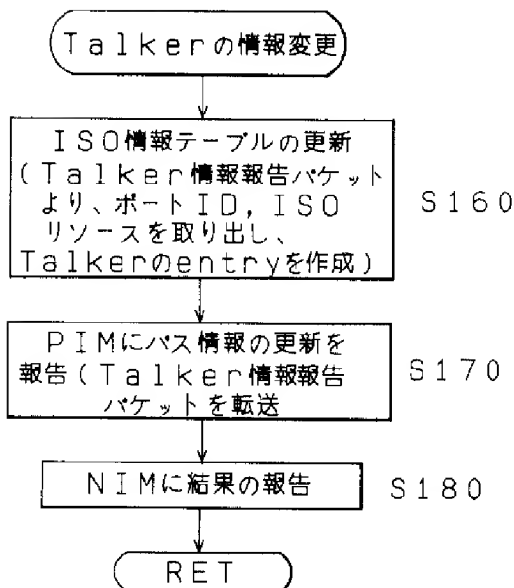
【図27】



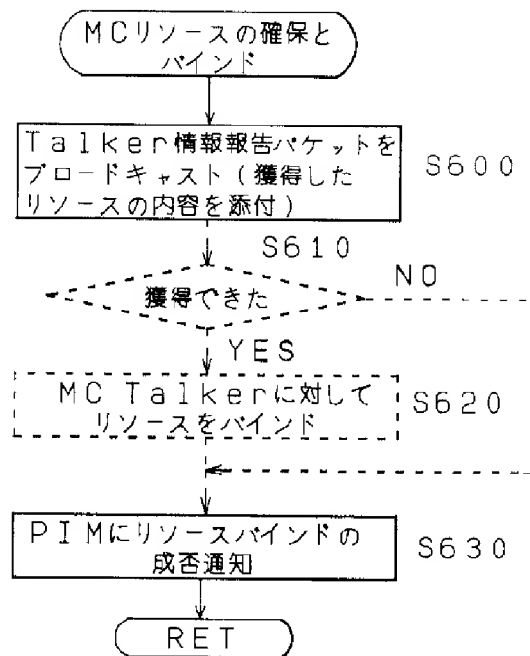
【図13】



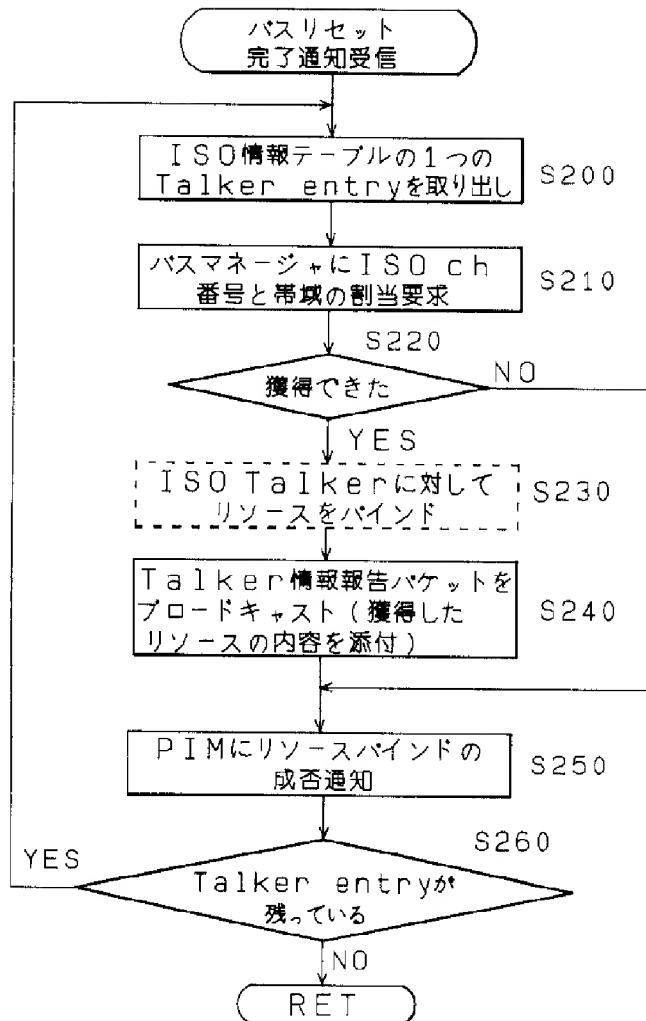
【図14】



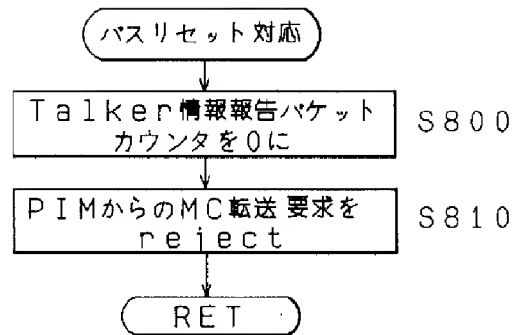
【図17】



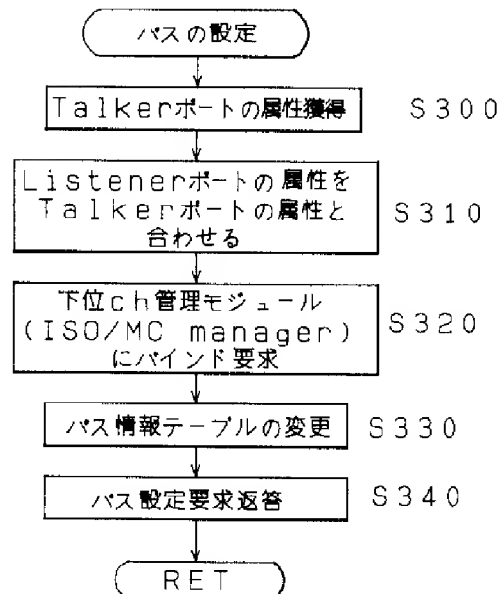
【図16】



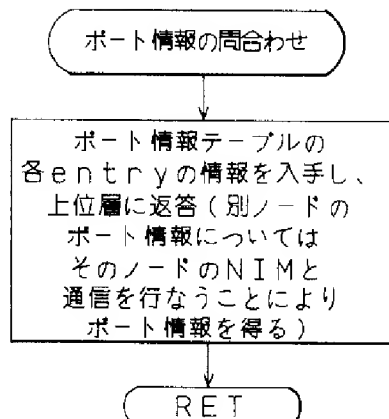
【図22】



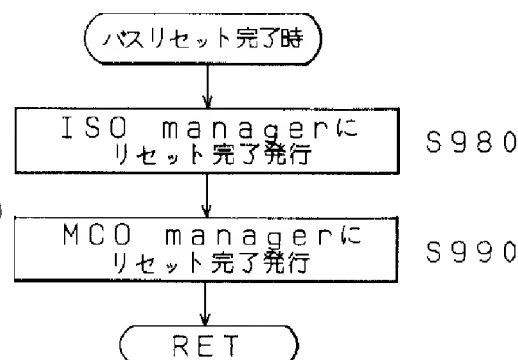
【図24】



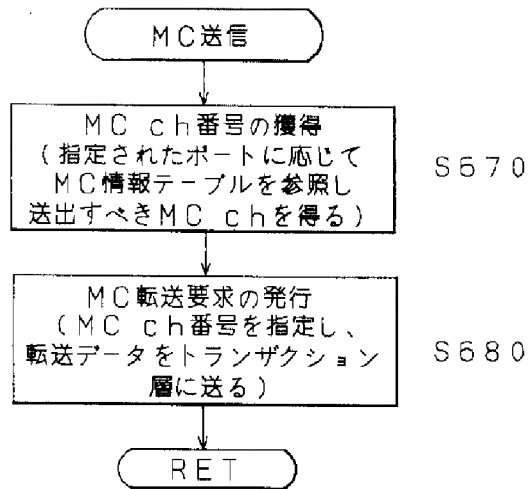
【図30】



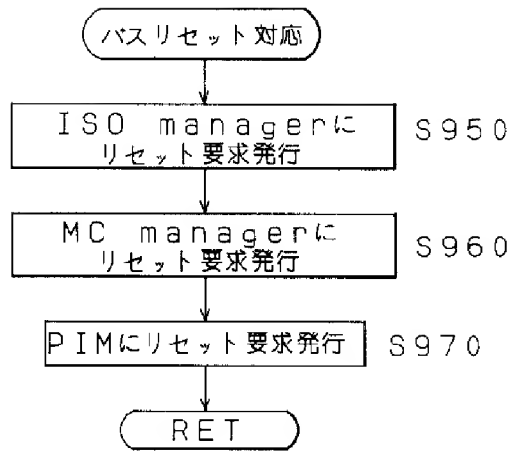
【図33】



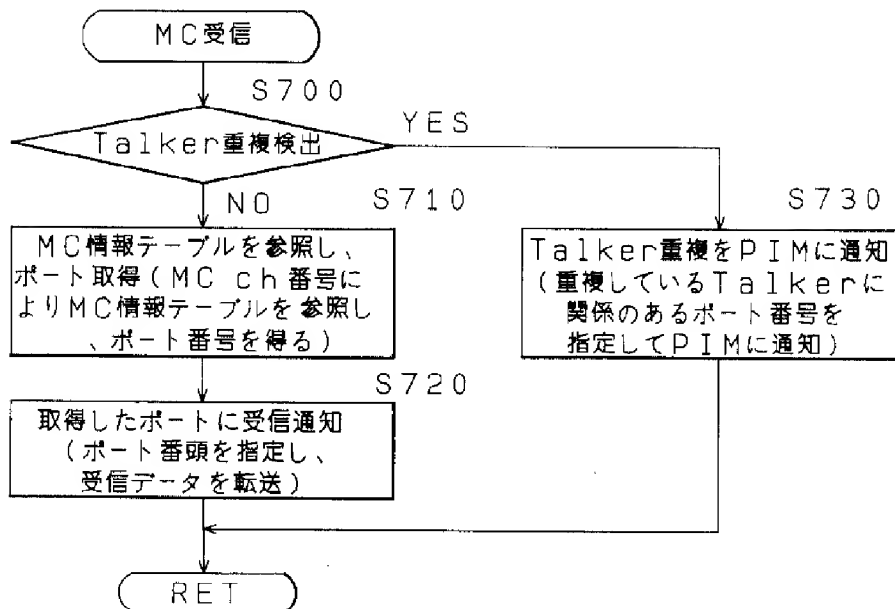
【図19】



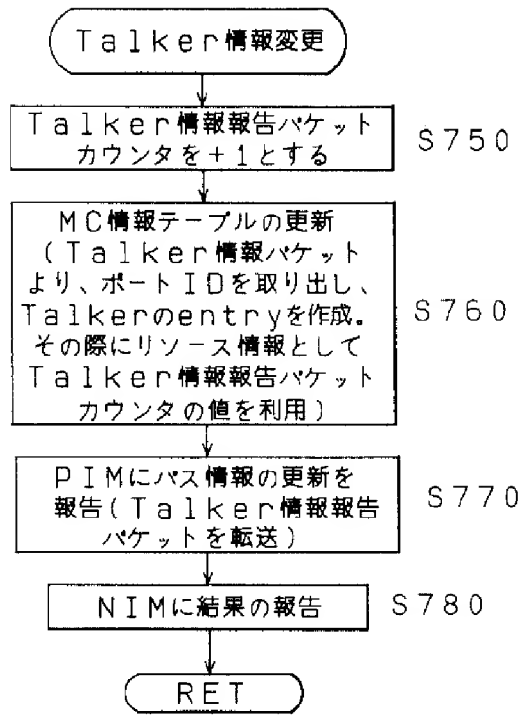
【図32】



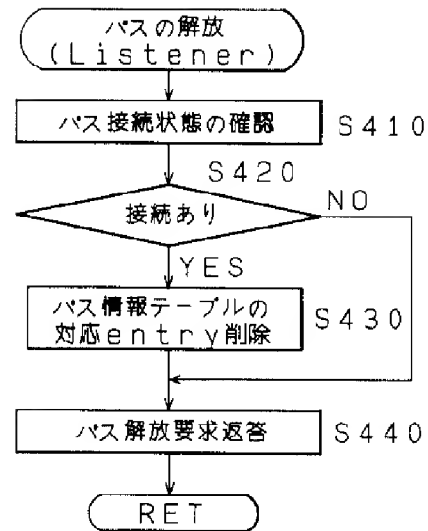
【図20】



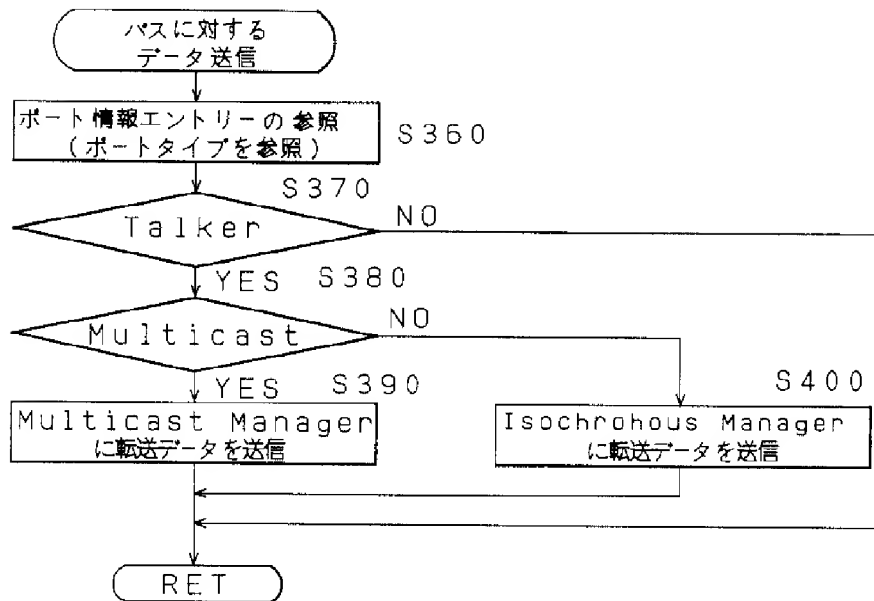
【図21】



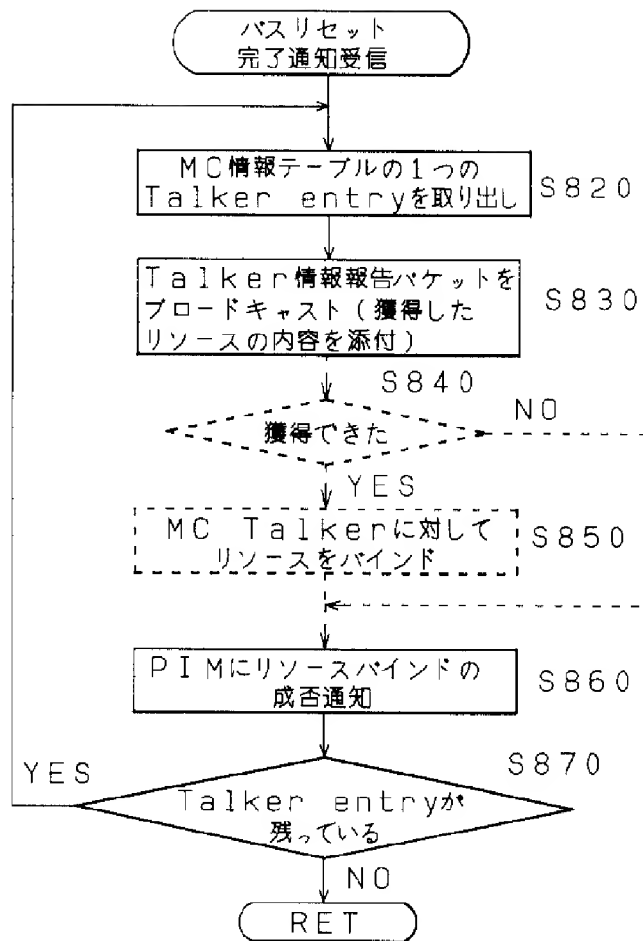
【図26】



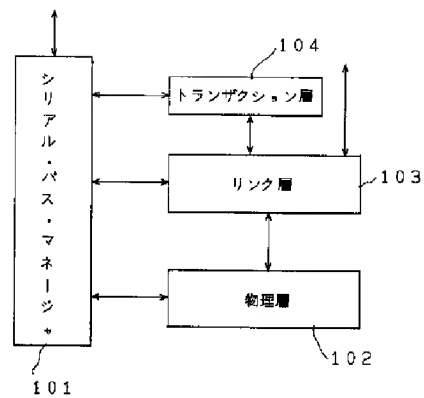
【図25】



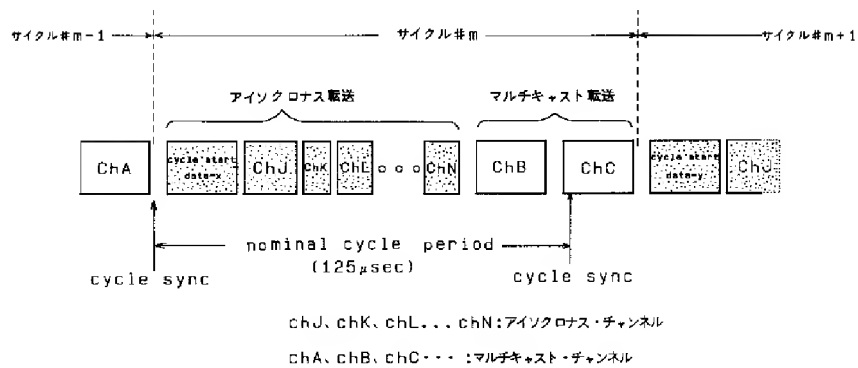
【図23】



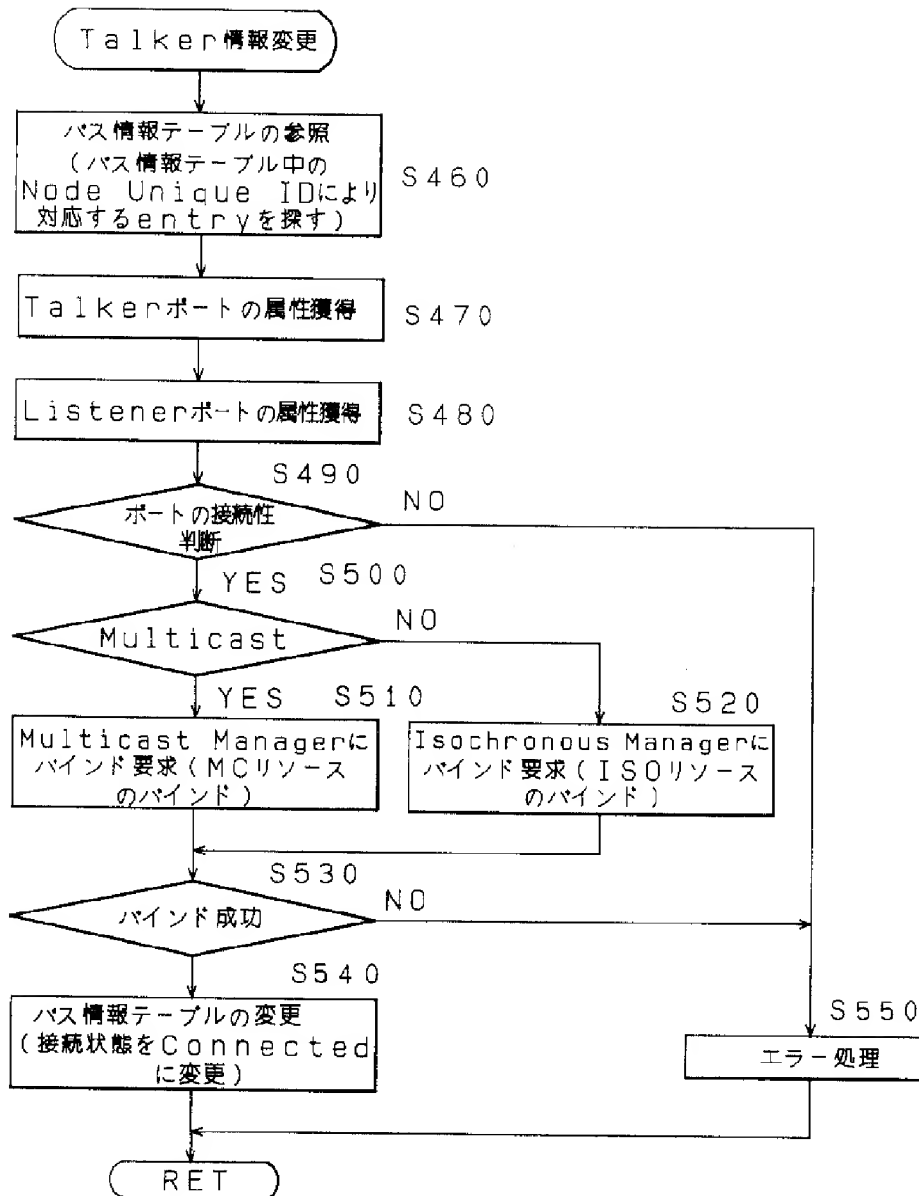
【図35】



【図34】



【図29】



【図31】

